Report Rapport

Atomic Energy Control Board
P.O. Box 1046
Ottawa, Canada
K1P 5S9

Commission de contrôle de l'énergie atomique
C.P. 1046
Ottawa, Canada
K1P 5S9

INFO-0145

COMPUTER MODELLING OF AN UNDERGROUND MINE VENTILATION SYSTEM

by

Senes Consultants Limited

A research report prepared
for the

Atomic Energy Control Board
Ottawa, Canada

RESEARCH REPORT

December 1984

RÉSUMÉ

Dans le milieu de travail de l'industrie minière de l'uranium, il est crucial de maintenir la concentration des produits de filiation du radon dans l'air à des niveaux acceptables. Puisque des modèles informatiques peuvent être utilisés pour concevoir les installations d'aérage d'une nouvelle mine et pour évaluer les effets des projets de modification à l'aérage des mines existantes, la Commission de contrôle de l'énergie atomique (CCEA) a entrepris la présente étude pour examiner les modèles d'aérage existants et pour mettre au point un modèle adapté aux besoins de ses spécialistes.

La présente étude a commencé par une revue de la documentation disponible sur les modèles informatiques d'aérage et, plus particulièrement, sur les modèles conçus pour prévoir les concentrations de produits de filiation du radon. Quoiqu'il existe des différences de détail entre les modèles, il appert que les méthodes fondamentales de calcul sont similaires pour tous les modèles. Un modèle élaboré par l'université Queen's a donc été retenu parce qu'il contenait la plupart des caractéristiques souhaitées et qu'il était disponible immédiatement.

Par la suite, le programme d'ordinateur de l'université Queen's (en réalité deux programmes, l'un traitant d'aérage de la mine, l'autre calculant les concentrations des produits de filiation du radon) a été élargi et mis à l'essai. Le présent rapport fournit la documentation pertinente pour préparer et utiliser ces modèles. Il décrit également la justification mathématique des méthodes de calcul utilisées.

ABSTRACT

The ability to control workplace short-lived radon daughter concentrations to appropriate levels is crucial to the underground mining of uranium ores. Recognizing that mine ventilation models can be used to design ventilation facilities in new mines and to evaluate proposed ventilation changes in existing mines the Atomic Energy Control Board (AECB) initiated this study to first investigate existing mine ventilation models and then develop a suitable model for use by AECB staff.

At the start of the study, available literature on mine ventilation models, in particular models suitable for the unique task of predicting radon daughter levels, were reviewed. While the details of the models varied, it was found that the basic calculation procedures used by the various models were similar. Consequently, a model developed at Queen's University that not only already incorporated most of the desired features but was also readily available, was selected for implementation.

Subsequently, the Queen's computer program (actually two programs, one for mine ventilation and one to calculate radon daughter levels) was extended and tested. The following report provides the relevant documentation for setting up and running the models. The mathematical basis for the calculational procedures used in the models are also described.

DISCLAIMER

The Atomic Energy Control Board is not responsible for the accuracy of the statements made or opinions expressed in this publication and neither the Board nor the authors assume liability with respect to any damage or loss incurred as a result of the use made of the information contained in this publication.

ACKNOWLEDGEMENTS

This report was prepared under the direction of Dr. L. M. Lowe of SENES Consultants Limited. Mr. D. Beals of SENES was instrumental in implementing and documenting the computer models described here. The basic computer models were originally developed at Queen's University under the direction of Dr. K. Notley. Dr. Notley's continuing help and advice are gratefully acknowledged. Finally, we also wish to acknowledge the input concerning user needs and thermodynamic considerations provided by Mr. P. Pullen, private consultant, and Hatch Associates Limited, respectively.

Table of Contents

Page

ABSTRACT i

ACKNOWLEDGEMENTS ii

1.0 INTRODUCTION 1-1

2.0 PRINCIPLES OF MINE VENTILATION ANALYSIS 2-1
- 2.1 The Analytical Approach 2-1
- 2.2 Basic Formulae 2-1
- 2.3 Basic Network Theory 2-2
- 2.4 An Example 2-3
- 2.5 The Hardy Cross Method 2-5
- 2.6 Chord and Mesh Selection 2-7
- 2.7 The Example Continued 2-7

3.0 BASIC CONCEPTS OF RADON AND RADON DAUGHTER LEVEL PREDICTION FOR URANIUM MINES 3-1
- 3.1 Radon Flux into the Mine Atmosphere 3-1
- 3.2 Growth of Working Levels 3-6
- 3.3 Age of Air Concept 3-9
- 3.4 A Worked Example 3-12
- 3.5 Recirculation Loops 3-13
- 3.6 Limitations of Approach 3-15
  - 3.6.1 Neglect of Physical Removal Mechanisms 3-15
  - 3.6.2 Approach to Mixing Air Flows 3-17

Table of Contents (continued)

Page

4.0 USER'S MANUAL FOR THE VENTILATION PROGRAM VENTPROG 4-1
- 4.1 Program Abstract 4-1
- 4.2 Model Inputs 4-1
  - 4.2.1 Terminal Inputs 4-1
  - 4.2.2.Branch Inputs (File 5) 4-2
  - 4.2.3 Fan Characteristic Curve Inputs (File 7) 4-4
- 4.3 Program Output 4-5
- 4.4 Two Worked Examples 4-7

5.0 USER'S MANUAL FOR THE RADON DAUGHTER LEVEL PROGRAM VENTRAD 5-1
- 5.1 Program Abstract 5-1
- 5.2 Model Inputs 5-2
  - 5.2.1 Branch File (File 7) 5-2
  - 5.2.2 Branch Emanation Characteristics (File 8) 5-2
- 5.3 Program Output 5-3
- 5.4 Two Worked Examples 5-3

REFERENCES R-1

APPENDIX A - TECHNICAL DESCRIPTION OF INDIVIDUAL MODULES FOR VENTPROG

APPENDIX B - TECHNICAL DESCRIPTION OF INDIVIDUAL MODULES FOR VENTRAD

APPENDIX C - PROGRAM LISTING - VENTPROG

APPENDIX D - PROGRAM LISTING - VENTRAD

APPENDIX E - REVIEW OF EXISTING MINE VENTILATION MODELS

List of Figures

Figure	Follows Page
2.1 Kirchoff's First Law	2-2
2.2 Kirchoff's Second Law	2-2
2.3a Ventilation Circuit of a Simple Mine	2-3
2.3b Equivalent Network Representation of a Mine Ventilation Circuit	2-3
2.4 Illustration of Trees, Tree Branches and Chords for Network Shown in Figure 2.3	2-3
2.5 Example Network with Meshes Used in Example Illustrated	2-4
3.1 Radon Sub-Series	3-1
3.2 Working Level Growth in Initially Pure Radon	3-6
3.3 Tunnel Concept	3-7
3.4 Two Interconnected Branches	3-9
3.5 Junction of Branches	3-11
3.6 The Worked Example	3-12

List of Figures (continued)

		Follows Page
3.7a	Simple Recirculation Loop	3-13
3.7b	Compound Recirculation Loop	3-13
3.8	Working Level-Flow Relationship in a Recirculation Loop	3-15
3.9	Effect of Deposition on Radon Daughter Ingrowth	3-16
4.1	Unsolvable Circuits	4-4
4.2	Fan Characteristic Curve	4-5
4.3	Network Diagram For Example 1	4-7
4.4	Network Diagram For Example 2	4-8
5.1	Network Diagram for Example 1	5-3
5.2	Network Diagram for Example 2	5-3

List of Tables

Table		Follows Page
2.1	Resistance, Flows and R|Q|Values for the Example Network	2-7
2.2	Tabular Listing of Flow Values in the Hardy Cross Method	2-9
3.1	Radon Diffusion Lengths in Various Materials	3-5
3.2	Radon Daughter Growth Curve Parameters	3-6
3.3	Branch Parameters and Preliminary Calculation for Worked Examples	3-12
4.1	Control Codes Entered at the Terminal	4-1
4.2	Branch Data File Content and Format (File 5)	4-3
4.3	The Five Branch Types	4-3
4.4	Fan Characteristic Curve Data File Content and Format	4-4
4.5	Branch File For Example 1	4-7
4.6	Fan Input File for Example 1	4-7

List of Tables (continued)

Follows Page

4.7 What Appears on the Screen for Example 1 — 4-8

4.8 Output File for Example 1 — 4-8

4.9 Branch Input File for Example 2 — 4-8

4.10 Fan Input File for Example 2 — 4-8

4.11 Output File for Example 2 — 4-8

5.1 Format and Content for Branch File for VENTRAD (File 7) — 5-2

5.2 Format and Content of Emanation Characteristics File for VENTRAD (File 8) — 5-2

5.3 Branch File for Example 1 — 5-3

5.4 Emanation Data File for Example 1 — 5-3

5.5 Output for Example 1 — 5-3

5.6 Branch Input File for Example 2 — 5-3

5.7 Emanation Characteristics File for Example 2 — 5-3

5.8 Output for Example 2 — 5-3

1.0 INTRODUCTION

The purpose of this study, sponsored by the Atomic Energy Control Board of Canada (AECB), was to develop a practical computer model for the calculation of ventilation parameters applicable to underground uranium mine environments. The computer model will be used by the AECB in various applications, for example as a tool to help evaluate ventilation proposals submitted by mining companies.

In the past, the AECB had been using their experience and general background knowledge to judge ventilation designs or changes to existing procedures proposed by the mining companies. However the ever-changing technology and increasing complexity of the mines now requires a more quantitative and perhaps more rigorous scientific analysis. Besides being put to use as an "audit" tool, the final model may also be used as a "what if" program to assess the effects of various postulated changes to existing ventilation networks. In this manner, potential beneficial changes to existing designs might be identified. It is expected that in many cases, only subsections of a complete mine ventilation system would be modelled.

One aim of the design of the computer model was to make the model modular in nature to accommodate contingencies and to facilitate program upgrading. The two-step approach suggested for use in this study, the calculation of conventional ventilation parameters folowed by the calculation of radiation parameters is consistent with the objective.

The following report provides documentation for two computer models, a mine ventilation program and a radon daughter level program. The programs are intended to be used in a two phase process whereby the balanced network solution from the ventilation model is used, along with branch radon emanation characteristics, in the radon daughter level program. The ultimate goal of the modelling process is to predict radon and radon daughter levels in a mine.

In order to develop the inputs required for the mine ventilation computer program, a mine network, that is, a mathematical representation of a real mine, must be developed. The inputs developed from this process are: branch node numbers, branch types (e.g., normal, regulated, fan, etc.), branch resistance or branch geometry and friction factor, preliminary flows and fan curves. The mine ventilation program will solve the circuit utilizing these inputs, yielding balanced flows, pressure drops and power requirements as output.

A data file containing the solution from the mine ventilation program complete with node numbers, balanced flow rates, branch lengths and cross sectional areas is part of the input for the radon daughter level program. Another file, with branch emanation characteristics and initial radon and radon daughter levels, is also required for the radon daughter level program. The program can also deal with filtration units. The output for the radon level program contains radon and radon daughter level predictions for each branch.

The report provides several levels of detail for the user. User's manuals for both programs are presented in Chapters 4 and 5 for the ventilation and radon level programs respectively, and are intended to provide a stand alone guide for use of the programs for the experienced user such as a mine engineer. A program abstract, input content and format, output interpretation and worked examples are provided in the user's manual. Chapter 2 and 3 detail the equations, theories and assumptions used to develop the ventilation and radon level programs respectively.

Appendices A and B provide descriptions of all of the subprograms, complete with variable dictionaries for both programs. Detailed listings of both programs are provided in Appendices C and D.

At the outset of the study, a general review of model concepts, and existing mine ventilation models and programs was undertaken and presented to the AECB in the form of a progress report. The findings of this review are summarized in Appendix E. In general, it was found that, while the details of operation, input/output options, and advantages and disadvantages of the models varied, the basic methodologies used in the models were similar. For this reason, a model developed at Queen's University that incorporated most of the desired model options and which was readily available to the study team was used as the starting point for the models developed in this study.

2.0 PRINCIPLES OF MINE VENTILATION ANALYSIS

2.1 The Analytical Approach

The formulae and computational methods used in the mine ventilation computer program are detailed in the following section.

The basic formulae used in the mine ventilation circuits are simple, however, the complexity of the system being analysed requires a rational step by step approach. This approach is provided by an analytical method called network analysis. The basic formulae will first be outlined in this chapter and then a general discussion of network theory will be undertaken in order to illustrate how these formulae apply to large mine ventilation circuits. A simple example will also be presented. No detailed derivations are given in this report. For a more comprehensive treatment of the topic, it is recommended that the reader refer to standard discussions such as those given in Hartman (1982).

2.2 Basic Formulae

Head loss in an airway is directly proportional to the square of the flow through it. Atkinson's equation expresses this relationship as:

$$H_L = RQ^2 \qquad (2.1)$$

where

H_L = head loss(Pa)

Q = flow(m^3/s)

R = resistance of an airway($N.s^2/m^8$)

R can be calculated from:

$$R = \frac{KP(L+L_e)}{A^3} \qquad (2.2)$$

where

K = *empirical friction* factor (Kg/m^3)

P = perimeter of duct (m)

L = length of duct (m)

L_e = equivalent length calculated from shock losses (Hartman 1982)(shock losses are defined as losses due to changing the direction of airflow or the area of a duct)

A = area of duct (m^2)

Kirchhoff's First Law states that the summation of flows at a junction is zero (see Figure 2.1). If a quantity of air leaving a node is defined as positive and the quantity of air entering a node as negative it follows that:

$$Q_1+Q_2-Q_3-Q_4 = 0 \qquad (2.3)$$

$$\text{or} \quad \Sigma Q=0$$

Kirchhoff's Second Law states that the summation of pressure drops around a closed loop must be equal to zero or (see figure 2.2):

$$\Sigma H_L = 0$$

$$\text{or } \Sigma H_L=H_{L1} + H_{L2} + H_{L3} - H_{4L} = 0 \qquad (2.4)$$

Note that H_{L4} is subtracted because the direction of flow is against the clockwise direction of the summation. Substituting Atkinson's equation into the above pressure drop summation:

$$\Sigma H_L = R_1 Q_1{}^2 + R_2 Q_2{}^2 + R_3 Q_3{}^2 - R_4 Q_4{}^2 = 0 \qquad (2.5)$$

2.3 Basic Network Theory

We now have the basic formulae required to analyse complex circuits. Network analysis offers a rational step-by-step method by which to approach these circuits. Before the approach is outlined some definitions are needed.

FIGURE 2·1

KIRCHHOFF'S FIRST LAW

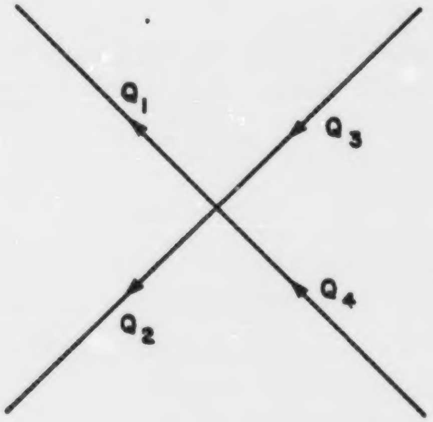
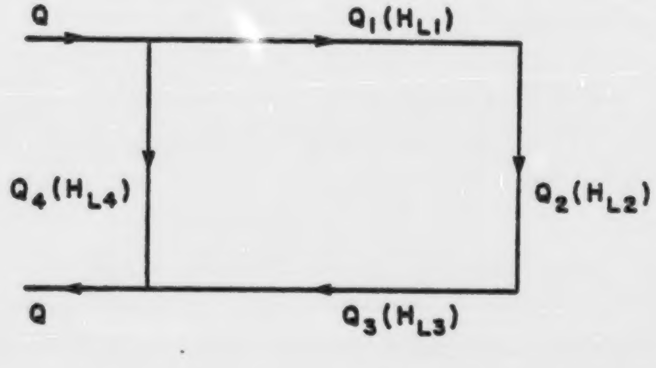


FIGURE 2·2

KIRCHHOFF'S SECOND LAW



Q
$Q_1(H_{L1})$
$Q_4(H_{L4})$
$Q_2(H_{L2})$
Q
$Q_3(H_{L3})$

A network is a mathematical representation of a real-world mine ventilation system. It is an assembly of data associated with the airways in the mine ventilation system. Figure 2.3a indicates the ventilation circuit of a simple mine and 2.3b shows a network representation of the mine. The airways in the system are called branches. The branches are interconnected at points called nodes. Each branch is represented by an initial node and a final node. Airflow direction is defined as positive if it goes from the initial to the final node and vice versa.

A network can be divided into a series of closed loops or meshes. A tree is a connected subnetwork that contains all of the nodes in the network, but no meshes. Figure 2.4 shows the example network divided into two different trees. The tree branches are the branches contained in a tree. The branches not contained in a tree are called chords. There is one chord per mesh in the network. If there are N_n nodes there are N_n-1 tree branches in a network. Since by definition the remaining branches are chords there are $m=N_b-(N_n-1)$ chords and meshes in a network where N_b is the number of branches.

2.4 An Example

With the foregoing basic definitions, it is possible to apply Atkinson's equation and Kirchhoff's laws to a network. Clearly, Atkinson's equation can be applied to each of the branches in the network yielding N_b linearly independent equations. A theorem of network analysis states that there are N_n-1 linearly independent node equations, (Kirchhoff's First Law) that can be applied to sum flows at the nodes of the network (Shearer 1971). Another network theorem states there are $m=N_b-N_n+1$ linearly independent mesh equations (Kirchhoff's Second Law) that sum pressures around the meshes of the network.

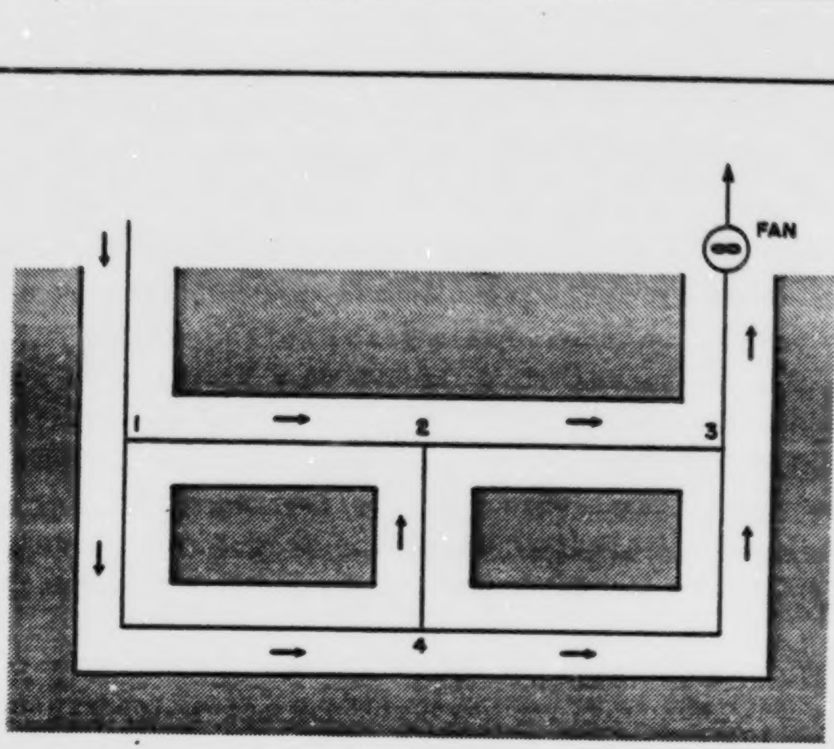


FIGURE 2·3a

VENTILATION CIRCUIT OF A SIMPLE MINE

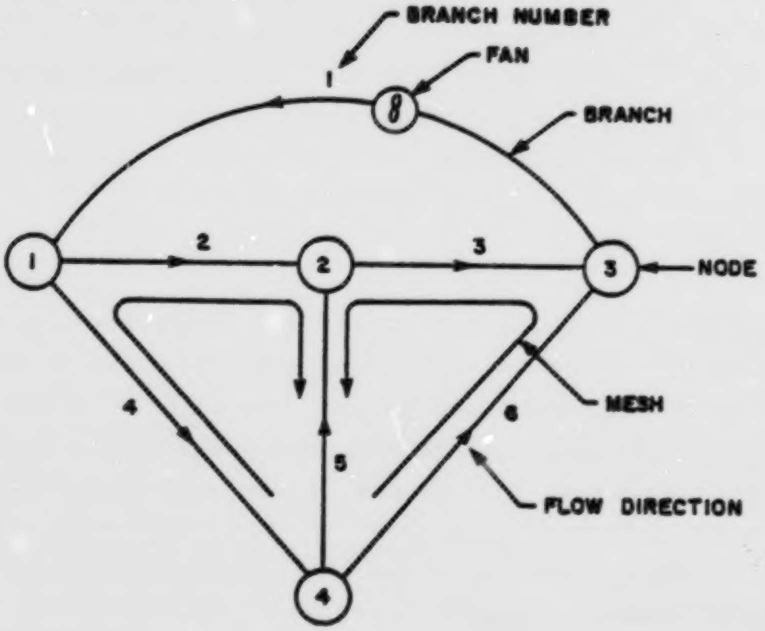
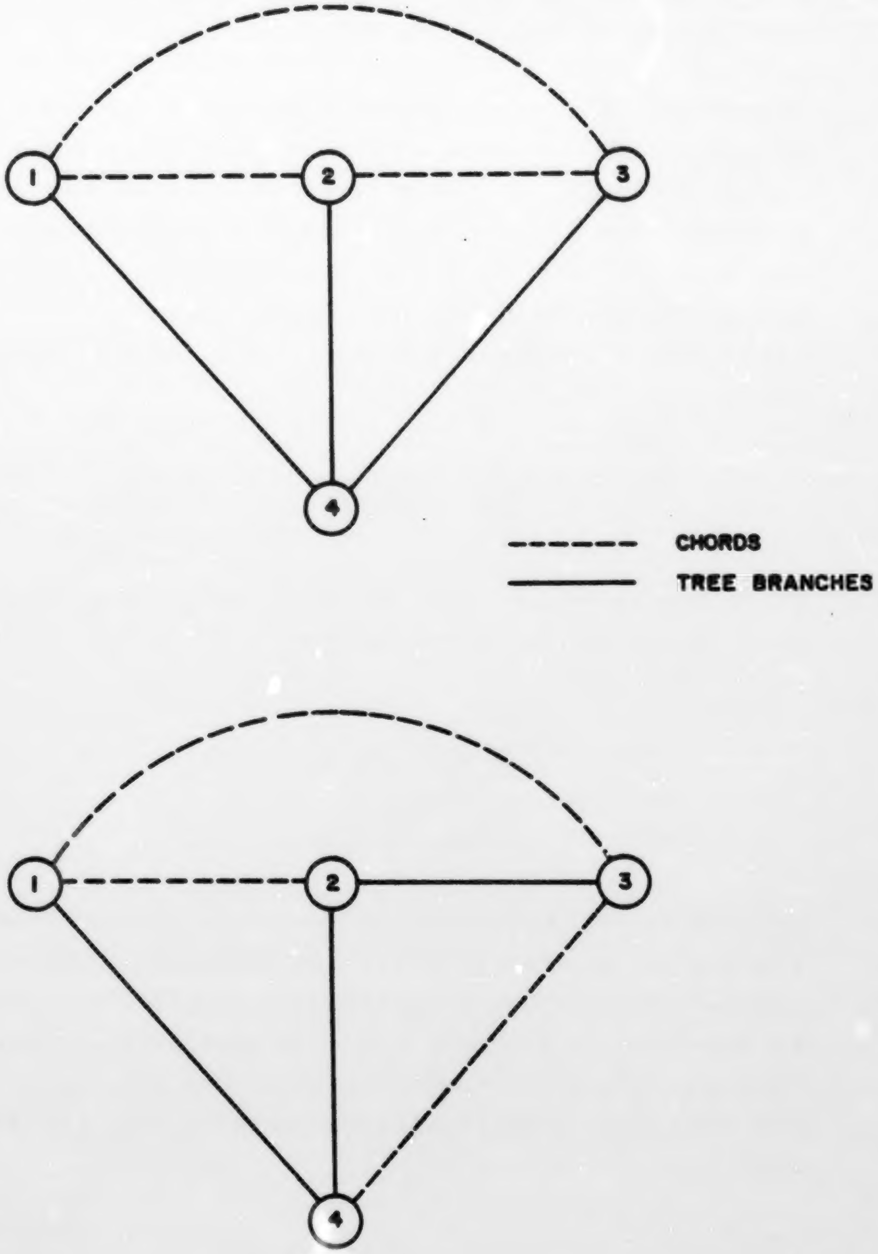

BRANCH NUMBER
1
FAN
BRANCH
2
3
1
2
3
NODE
MESH
4
5
6
4
FLOW DIRECTION

FIGURE 2·3b

EQUIVALENT NETWORK REPRESENTATION OF A MINE VENTILATION CIRCUIT

FIGURE 2·4

ILLUSTRATION OF TREES, FREE BRANCHES AND CHORDS FOR NETWORK SHOWN IN FIGURE 2·3


1
2
3
4
CHORDS
TREE BRANCHES
1
2
3
4

An example adapted from Hartman (1982) will help illustrate these concepts. Consider the network in Figure 2.3b. For this network:

Number of branches = N_b = 6

Number of nodes = N_n = 4

Therefore, there are three tree branches and three chords and meshes. Let us assume at the present that the known quantities for the network are the airway resistances and the static head of the fan. The following quantities are unknown: (1) head loss for each branch; (2) direction and quantity of airflow for each branch. Since there are 6 branches there are 12 unknowns. Since there are 6 branches there exist 6 formulations of Atkinson's equation:

$$H_{L1} = R_1\,Q_1^2 \qquad H_{L4} = R_4\,Q_4^2$$
$$H_{L2} = R_2\,Q_2^2 \qquad H_{L5} = R_5\,Q_5^2$$
$$H_{L3} = R_3\,Q_3^2 \qquad H_{L6} = R_6\,Q_6^2$$

There are as stated earlier, N_n-1 independent node equations. We will arbitrarily choose nodes 1,2 and 3. Therefore the node equations are:

Node 1: $-Q_1+Q_2+Q_4 = 0$

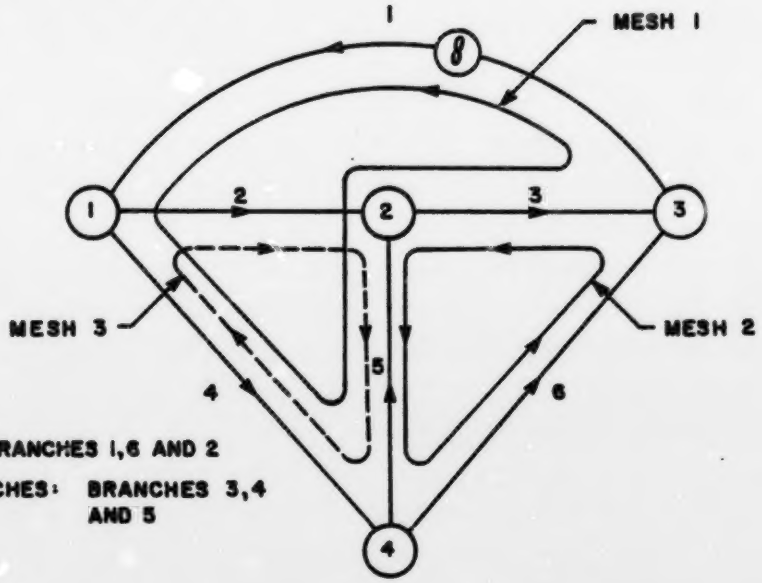
Node 2: $-Q_2+Q_3-Q_5 = 0$

Node 3: $Q_1-Q_3-Q_6 = 0$

An iterative solution called the Hardy Cross method will eventually be used to solve the problem. A method used to choose the meshes in order to optimize the speed of this solution will be detailed in section 2.6. The chords and meshes chosen using this optimization technique are the ones shown in Figure 2.5. The equations summing the pressure drops for the three meshes are:

FIGURE 2·5

EXAMPLE NETWORK WITH MESHES USED IN EXAMPLE ILLUSTRATED


1
8
MESH 1
1
2
2
3
3
MESH 3
MESH 2
5
4
6
4

CHORDS: BRANCHES 1,6 AND 2

TREE BRANCHES: BRANCHES 3,4 AND 5

Mesh 1: $H_1+H_4+H_5+H_3-H_f=0$
(H_f = fan pressure)
Mesh 2: $H_6-H_3-H_5=0$
Mesh 3: $H_2-H_5-H_4=0$

Note that the direction of flow in each of the chords determine the direction of positive flow in a mesh.

One can begin to substitute Atkinson's formulation into the path equations and the Q's from the node equations into Atkinson's equations, however the equations are nonlinear and an exact solution is impossible. An iterative technique called the Hardy Cross method is used to provide a reasonable approximation and is decribed in the next section.

2.5 The Hardy Cross Method

The following description of the Hardy Cross method was partially taken from Hartman (1982):

Step 1

Preliminary flows are assumed for all branches.

Step 2

A correction factor ΔQ is developed for each mesh (for i = 1,2...m where m is the number of meshes) using equation 2.6 below.

$$a)\ \Delta Q_i = -\frac{\sum_{j=1}^{n_{bi}} b_{ij}\,(R_j\,|Q_j|\,Q_j-h_{nj}-h_{fj})}{\sum_{j=1}^{n_{bi}} b_{ij}^2\,[2R_j\,|Q_j| - s_{fj}]} \qquad (2.6)$$

where

n_{bi} = number of branches in the ith mesh

h_{nj} = natural ventilation pressure (Pa) in the jth branch of the ith mesh

h_{fj} = fan head (Pa) in the jth branch of the ith mesh

s_{fj} = slope of fan characteristic curve in the jth branch of the ith mesh (fan curves will be discussed more fully in section 4.2.3)

ΔQi = Hardy Cross correction factor for the ith mesh

b_{ij} = 1 if branch j is contained in mesh i and has same flow direction as the flow in the chord in the mesh

b_{ij} = -1 if branch j is contained in mesh i and has the opposite flow direction as the flow in the chord in the mesh

b_{ij} = 0 if branch j is not contained in mesh i

Step 3

The correction factor for the ith mesh is then added to each of the branch flows in that mesh.

$$Q_j \leftarrow Q_j + b_{ij}\Delta Q_i \text{ for } i = 1, 2----, n_{bm} \qquad (2.7)$$

Step 4

If the predetermined limiting number of iterations (where one iteration involves varying i from 1 to m) has been performed for Step 2, stop. When this occurs, it generally means that convergence has not occurred. In the computer program, 30 iterations are recommended.

Step 5

If $\Delta Q_i \leq E_v$, the predetermined error value, for all of the meshes in Step 2, the solution has been achieved. A value of $E_v = 0.005\ m^3/s$ is recommended for use in the computer program.

2.6 Chord and Mesh Selection

In order to optimize the rate of convergence of the Hardy Cross method, the following procedure is used in the computer program to select the chords and meshes:

1. Fixed and then regulated flow branches are first chosen as chords.
2. If m chords have not been accumulated, where m is the number of meshes, fan branches are then chosen as chords.
3. If m chords have still not been accumulated, unrestricted flow branches with higher values of $R\ |Q|$ are chosen until there are m chords.
4. Now that all chords have been chosen, the meshes are found. Only one chord is allowed per mesh.

2.7 The Example Continued

Let us continue to look at the example outlined in section 2.4 (Figure 2.5) and apply the Hardy Cross method. The resistances and initial flows and subsequent $R\,|Q|$ values of the branches are listed in Table 2.1. The fan head in this simplified problem is a constant 1990.7 Pa. Hence, the slope s_{fj} referred to in step 2 of the Hardy Cross method is zero.

It can be seen that, in descending order of $R\,|Q|$ the branch order is 1>6>3>2>4>5. From the chord selection procedure detailed in the previous section, we know that the fan branch is the first

Table 2.1

Resistance, Flows and R |Q| Values for the Example Network

Branch Number	Resistance (R) (N.s^3/m^8)	Preliminary Flow (Q) (m^3/s)	R |Q|
1	0.581	46.4	27.0
2	0.475	21.7	10.3
3	0.849	23.6	20.0
4	0.363	24.7	9.0
5	1.056	1.9	2.0
6	0.916	22.8	20.9

chord. We also know that the next two highest $R|Q|$ branches, 6 and 3, should be chosen as chords. Since three meshes, with one chord per mesh, cannot be formed with these chords, one of the three chords must be changed. Since branch 3 is the lowest, non-fan $R|Q|$ chord it is dropped and the next highest $R|Q|$ branch, 2, is chosen.

Using branches 1, 6 and 2 as chords, the only three meshes that can be chosen are shown in Figure 2.5. We are now ready to apply the Hardy Cross method with the iterations ceasing when $\Delta Q<0.1$ for all meshes. (The value of $\Delta Q<0.1\ m^3/$ was used for demonstration purposes in this example the actual program uses a smaller value.)

Step 1

We already have the preliminary flows listed in Table 2.1 that are required in step 1 of the Hardy Cross method.

Steps 2 and 3

Mesh 1:

$$\begin{aligned}\Delta Q_1 &= -\frac{R_1|Q_1|Q_1 + R_4|Q_4|Q_4 + R_5|Q_5|Q_5 + R_3|Q_3|Q_3 - H_F}{2[R_1|Q_1| + R_4|Q_4| + R_5|Q_5| + R_3|Q_3|]}\\ &= -\frac{(0.581)(46.4)^2+(0.363)(24.7)^2+(1.056)(1.9)^2+(0.849)(23.6)^2-1990.7}{2[(0.581)(46.4)+(0.363)(24.7)+(1.056)(1.9)+(0.849)(23.6)]}\\ &= 0.36\end{aligned}$$

$Q_1 \leftarrow Q_1 + \Delta Q_1 = 46.4 + 0.36 = 46.8$

$Q_4 \leftarrow Q_4 + \Delta Q_1 = 24.7 + 0.36 = 25.1$

$Q_5 \leftarrow Q_5 + \Delta Q_1 = 1.9 + 0.36 = 2.3$

$Q_3 \leftarrow Q_3 + \Delta Q_1 = 23.6 + 0.36 = 24.0$

Table 2.2

Listing of Flow Values in the Hardy Cross Method

		Branch Flows (m^3/s)						
Iteration	Mesh i	ΔQ_i	Q_1	Q_2	Q_3	Q_4	Q_5	Q_6
1	1	0.36	46.8		24.0	25.1	2.3	
	2	0.21			23.8		2.1	23.0
	3	0.22		21.9		24.9	1.9	
2	1	0.07	46.9		23.9	25.0	2.0	
	2	0.05			23.8		1.9	23.1
	3	0.07		21.9		24.9	1.8	

Mesh 2:

$$\Delta Q_2 = -\frac{R_6 |Q_6| Q_6 - R_3 |Q_3| Q_3 - R_5 |Q_5| Q_5}{2[R_6 |Q_6| + R_3 |Q_3| + R_5 |Q_5|]}$$

$$= -\frac{(0.916)(22.8)^2 - (0.849)(24.0)^2 - (1.056)(2.3)^2}{2[(0.916)(22.8) + (0.849)(24.0) + (1.056)(2.3)]}$$

$$= 0.210$$

$Q_6 \leftarrow Q_6 + \Delta Q_2 = 22.8 + 0.210 = 23.0$

$Q_3 \leftarrow Q_3 - \Delta Q_2 = 24.0 - 0.210 = 23.8$

$Q_5 \leftarrow Q_5 - \Delta Q_2 = 2.3 - 0.210 = 2.1$

Mesh 3:

$$\Delta Q_3 = -\frac{R_2 |Q_2| Q_2 - R_5 |Q_5| Q_5 - R_4 |Q_4| Q_4}{2[R_2 |Q_2| + R_5 |Q_5| + R_4 |Q_4|]}$$

$$= -\frac{(0.475)(21.7)^2 - (1.056)(2.1)^2 - (0.363)(25.1)^2}{2[(0.475)(21.7) + (1.056)(2.1) + (0.363)(25.1)]}$$

$$= 0.22$$

$Q_2 \leftarrow Q_2 + \Delta Q_3 = 21.7 + 0.2 = 21.9$

$Q_5 \leftarrow Q_5 + \Delta Q_3 = 2.1 - 0.2 = 1.9$

$Q_4 \leftarrow Q_4 + \Delta Q_3 = 25.1 - 0.2 = 24.9$

Step 4

Since the ΔQ values are greater than 0.1 m^3/s, go back to step 2. If the ΔQ values are all less than 0.1 m^3/s, the solution has been reached. It can be seen in Table 2.2 that, in this idealized example, only two iterations were required to get the ΔQ's below 0.1 m^3/s.

The above simplified example was also run in the computer program. A full description of the input and output for this example is contained in the first worked example in section in 4.4.

3.0 BASIC CONCEPTS OF RADON AND RADON DAUGHTER LEVEL PREDICTION FOR URANIUM MINES

Radon-222 is produced by the radioactive decay of radium-226, itself a decay product of uranium. Radium has a long half-life of 1622 years and thus acts as an effectively constant source of radon which has a relatively short half-life of 3.82 days. Radon is the only gaseous element in the uranium series and, on reaching this stage in the decay chain, radioactive elements have potential to escape from the host rock and enter the mine atmosphere.

The decay of radon into its next four successive daughter products is illustrated in Figure 3.1. These short-lived radioactive daughter products may become trapped in the human respiratory system where they constitute a health hazard due to the ionizing alpha radiation associated with their decay.

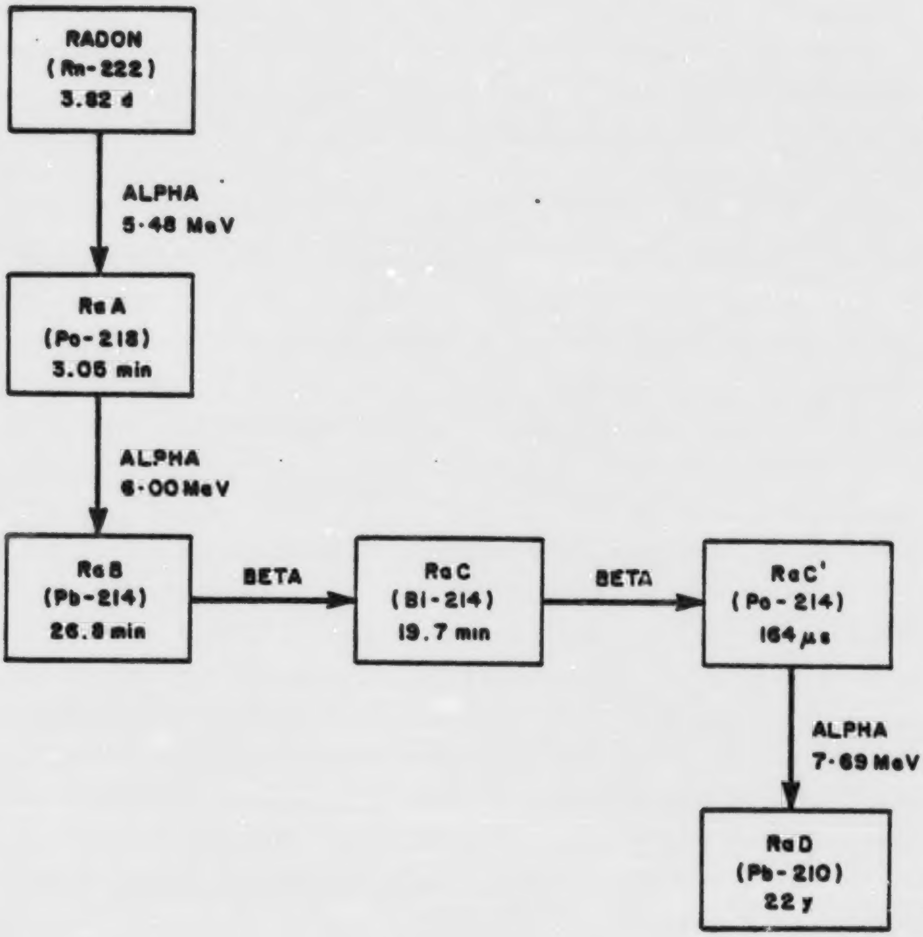
Concentrations of the short-lived radon daughter products are measured in working levels (WL). One working level is defined as any combination of the short-lived decay products of radon in 1 litre of air that will result in the ultimate emission by them of 1.3×10^5 MeV of alpha energy. Exposure to radon daughters is expressed as the product of working levels and time, the recognized unit being the working level month (WLM). One WLM results from exposure to a radon daughter concentration of one WL for one working month (170 hours). The Atomic Energy Control Board has specified a maximum permissible annual occupational exposure of 4 WLM.

3.1 Radon Flux Into The Mine Atmosphere

The migration of radon gas through porous media is influenced by many factors as discussed in the radon literature (e.g. Tanner

FIGURE 3-1

RADON SUB-SERIES


RADON
(Rn-222)
3.82 d
ALPHA
5.48 MeV
RaA
(Po-218)
3.05 min
ALPHA
6.00 MeV
RaB
(Pb-214)
26.8 min
BETA
RaC
(Bi-214)
19.7 min
BETA
RaC'
(Po-214)
164 μs
ALPHA
7.69 MeV
RaD
(Pb-210)
22 y

1964, 1978). The flux of radon from ore surfaces into the mine atmosphere is determined by the rate of production of diffusing radon and the rate at which interstitial radon migrates to exterior surfaces.

The radon source strength is in turn determined by the rate at which radon is produced through the decay of radium and by the fraction (often termed "emanation coefficient") of the radon which escapes the rock matrix and is free to diffuse in the pore spaces. Emanation coefficients ranging from 0.01 to 0.91 have been reported in the literature and depend on such factors as mineralogy, porosity and ore grade (Austin and Droullard 1978). A "typical" emanation coefficient of 0.2 has often been used to model radon sources from uranium mines.

The radon production rate (q) in a porous radium-bearing material can be expressed as:

$$q = [Ra] \times \rho \times E/P \times \lambda \quad \text{(Bq/s per m}^3\text{ of pore space)} \qquad (3.1)$$

$$= \frac{\beta max \times E}{P} = \frac{\beta}{P}$$

where

[Ra] = radium-226 concentration (Bq/g)

ρ = bulk density (g/m^3)

E = emanation coefficient

P = porosity (ratio of pore volume to total volume)

λ = radon decay constant ($2.1 \times 10^{-6}/s$)

β = emanating power ($Bq/s.m^3$)

βmax is the emanating power assuming all available radon is released to the pores.

A good approximation of the emanating power of run-of-mine material may be obtained by measuring the radon release rate of core-sized samples. At the Denison uranium mine, the average emanating power of such samples has been found to be around 0.85 $Bq/(s.m^3)$, although measured values have ranged from 0.1 to 1.3 $Bq/(s.m^3)$. The average emanation coefficient has been calculated to be about 0.02

Where radon is diffusing through a non-radioactive, porous body (e.g. waste rock, tailings cover material, etc.) the rate of diffusion from a plane source is given by (e.g. Tanner 1978)

$$C = C_o \exp \left[-z\sqrt{\frac{\lambda P}{D}}\right] \qquad (3.2)$$

where

z = perpendicular distance (m) from the source

C = radon concentration at distance z

C_o = radon concentration in the plane source

D = bulk diffusion coefficient (m^2/s)

The diffusion length L, the distance in which the concentration decreases by a factor of 1/e, is given by:

$$L = \sqrt{\frac{D}{\lambda P}} \qquad (3.3)$$

In a radioactive medium such as an ore body, the migration equations required modification. Applications of Fick's first law of diffusion and the steady state continuity equation lead to the solution (e.g. Tanner 1978):

$$C = C_o \left[(1 - \exp(-z/L)\right] \qquad (3.4)$$

where

$C_o = q/\lambda$, the radon concentration at z = infinity (i.e. deep within the source), and

= 0 at z = 0 (i.e. the radon concentration at the surface of the source)

At the air-surface interface, the radon flux is given by

$$J = -D \frac{\delta C}{\delta z} = -\beta x\ L \qquad (3.5)$$

In this situation, the controlling factors are the radium concentration, bulk density, emanating fraction, porosity and the bulk diffusion coefficient.

The bulk diffusion coefficient in water is almost four orders of magnitude smaller than in air. Hence the diffusion length L in water would be two orders of magnitude smaller than in air. It is then apparent that moisture content is very important in determining the rate of radon migration in undisturbed or disturbed ore. Temperature will also influence migration since, under freezing conditions, the radon will be distributed between the liquid and solid phases. Similarly, various factors affect the transport of fluids in the pore space such as air pressure cycles which will lead to convective motion within the rock thus modifying the rate of radon transport.

The proportion of radon in the mine atmosphere contributed by pure diffusion or water transport varies among mines.

Radon and radium may become dissolved or leached by mine water. The efficiency of this process depends on the water's chemical composition, temperature and pressure, as well as on the rock's porosity. Water transportation will then occur, until the water is released into the mine, where a high portion of radon may be emanated into the air. The degree of radon emanation from the mine water will be influenced by the radon concentration in the water, as well as by the degree of water agitation where it comes into contact with the mine atmosphere.

Several examples of the diffusion lengths have been compiled and some are given in Table 3.1 (adopted from Chambers et al 1980). Measured emanation rates from the walls of Denison ore have been found to lie between 0.074 and 0.44 Bq/(s.m^2) (Wheeler 1980). Diffusion lengths up of 0.3 m have been calculated.

As the emanating power and porosity of material can be measured, estimates of diffused radon gas released into mine air are possible. However in practice, the radon diffusion rate is known to depend on a very large number of different parameters, so accurate predictions are very difficult to make.

Aside from diffusion and transport directly from the rock walls, there are numerous other sources of radon in an underground mine atmosphere, some of which may be very substantial. Blasting will cause radon to be released immediately on rock fracture, and will also affect radon influx due to stress redistribution. Piles of loose muck will emanate radon, and mucking activities greatly enhance radon production.

Hydraulic emplaced fill, derived from uranium mill tailings, may emanate large quantities of radon, during placement and after hardening. The amount of radon released depends upon the composition and classification of the material used.

Dead ends with no ventilation may build up high concentrations of radon. In mines, bulkheads are usually employed to seal off such areas. However, it has proven very difficult to make bulkheads completely impervious to leakage. The leakage of "old" air with high radon daughter concentrations is a source of potential high exposures to workers near the source of leakage (See also Section 3.5 on recirculation loops).

Several researchers have published practical methods to estimate radon exhalation rates based on the theory discussed above. Calculations have been carried out for uranium mines in Elliot Lake (Archibald and Nantel 1984; Wheeler 1982), Saskatchewan (Cheng 1981) and the United States (Nielson et al 1979).

3.2 Growth of Working Levels

To calculate the growth of working levels in initially pure radon, decay product concentrations must be calculated for different air 'ages'. This involves determination of the number of atoms of each daughter product per unit volume, along with their corresponding activities. Such calculations, as described by Evans (1969), give rise to the characteristic working level growth curve shown in Figure 3.2. This figure assumes a constant source of radon, a good approximation considering that the half-life of radon is 3.82 days and the radon daughters reach equilibrium in 3 to 4 hours.

Using regression analysis, the growth curve was approximated over three time intervals by an equation of the form:

$$WL = KC_o t^y \qquad (3.6)$$

with the time increments and related regression parameters shown in Table 3.2. The parameter C_c is the radon concentration (Bq/m^3) and k is a proportionality constant (WL per 3700 Bq/m^3). The time ranges have been chosen so that the regression equations

Table 3.1

Radon Diffusion Lengths in Various Materials

Source	Diffusion Length (m)	Description
Published data (Tanner 1964)	Range of 2×10^{-30} to 2	Range of diffusion lengths of solid rock material to air
Mine A (Canada)	3	Quartz conglomerate with cracks and microfractures
Mine B (Canada)	0.02	Competent schist
Mine C (U.S.A.)	1.2	Porous sandstone
Mine D (Australia)	0.7	Protore piles, loose, 0.015% U_3O_8
(Australia)	0.6	Ore pile, loose, 0.44% U_3O_8
(Australia)	2.2	Possibility in broken ore pile
Model Mine, Wyoming	1.4	Dry sedimentary sandstone, 0.11% U_3O_8, 20-25 % porosity

FIGURE 3·2

WORKING LEVEL GROWTH
IN INITIALLY PURE RADON

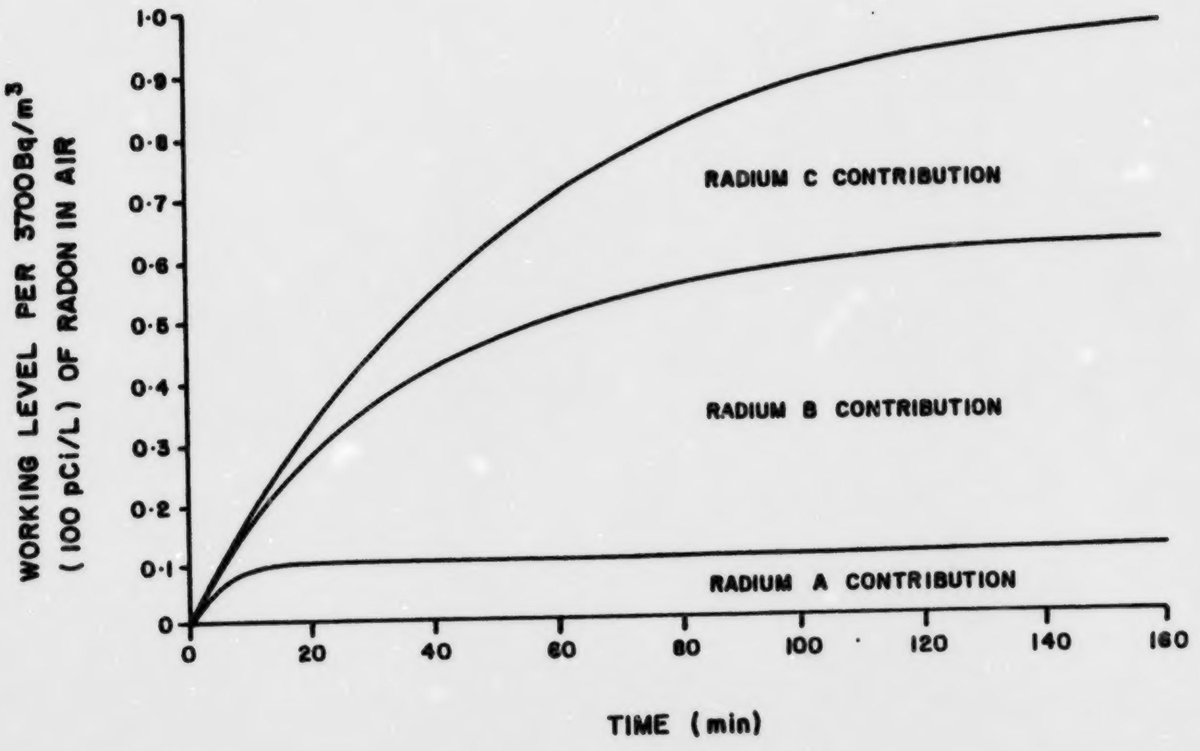


Table 3.2

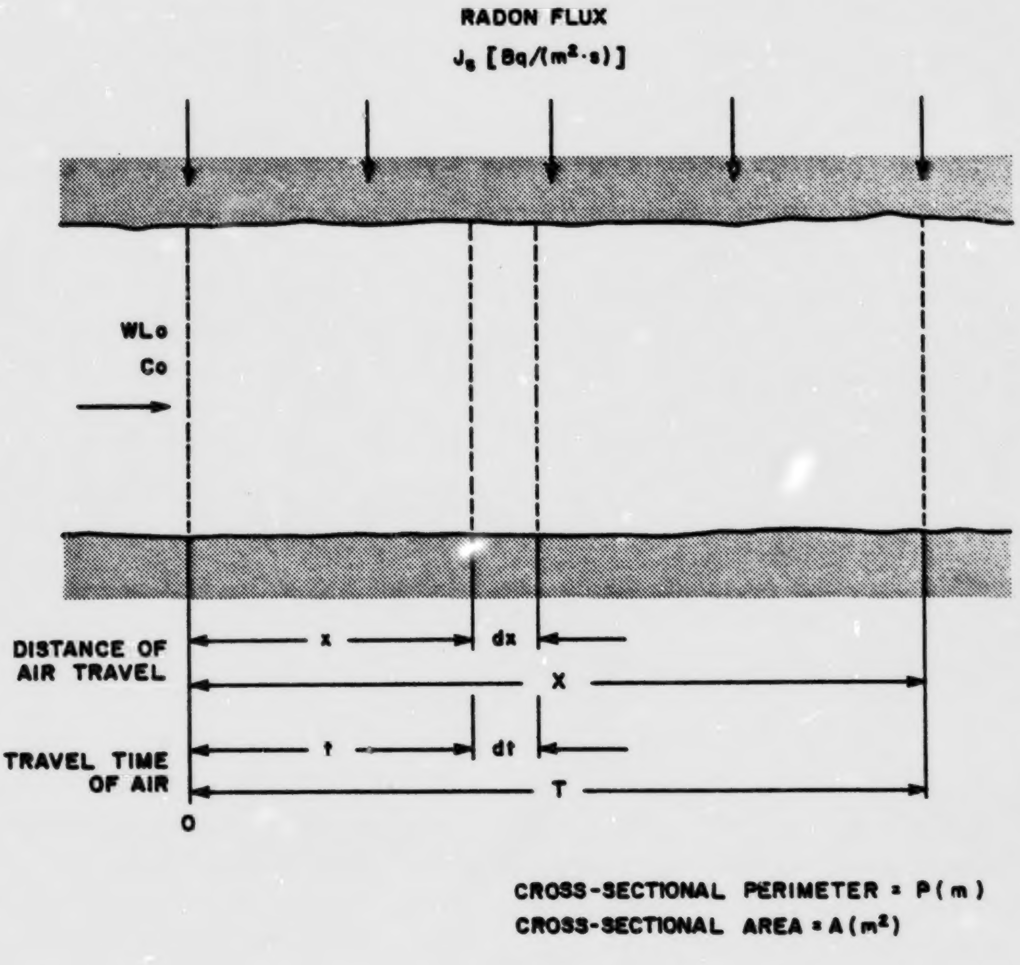
Radon Daughter Growth Curve Parameters

K	y	Time Range (min)	Equilibrium Ratio (E_r)
6.15×10^{-6}	0.86	0.0 - 41.45	0.0 - 0.55
23.15×10^{-6}	0.50	41.45 - 112.0	0.55 - 0.91
137.21×10^{-6}	0.12	112.0 - 220.0	0.91 - 0.99

Regression Equation: $WL = KC_o t^y$

or $E_r = \frac{WL}{C_o} = Kt^y$

FIGURE 3-3
TUNNEL CONCEPT


RADON FLUX
J_s [Bq/(m²·s)]
WLo
Co
DISTANCE OF AIR TRAVEL
x
dx
X
TRAVEL TIME OF AIR
t
dt
T
0
CROSS-SECTIONAL PERIMETER = P(m)
CROSS-SECTIONAL AREA = A(m²)

yield equal values where they overlap i.e. at t = 41.45 and 112.0 min. The equations approximate the time growth curve to within 5%. The equilibrium ratio values listed in the table.

The radon daughter concentration in a given airway consists of two components; 1) radon daughters resulting from decay of radon from previous airways (WL_1); and 2) radon daughters resulting from the decay of radon originating in that particular airway (WL_2).

Let us first consider the growth of WL_2 as described by Schroeder and Evans (1969). Assuming a uniform emanation rate over all the wall surfaces of a tunnel, the diffusion of radon into air travelling along such a tunnel may be depicted as shown in Fig. 3.3. The incremental radon concentration, dC, added to a small element of air travelling down the tunnel, in a time dt will be given by:

$$dC = \frac{J_s\ (Pdx)dt}{Adx} \qquad (3.7)$$

where

P = perimeter of the walls
A = cross-sectional area
Pdx= rock surface area of tunnel
Adx= volume of air element.

The working level contribution due to this radon concentration may be obtained by combining Eq. (3.6) with Eq. (3.7). By integrating this contribution for the whole travel time of air in the tunnel, the ultimate radon daughter concentration WL_2, due to diffused radon, will be given by:

$$WL_2 = \int_0^T \frac{J_s\ P\ K\ t^Y\ dt}{A}$$

or $$WL_2 = \frac{J_s\ P\ K\ (T^{(y+1)} - b)}{(y+1)\ A} \quad (3.8)$$

where

b = integration constant
= 0 for 0 < T ≤41.45 min.
= 0.047 for 41.45 < T ≤112.0 min.
= 0.420 for 112.0 < T ≤220 min.

and where the travel time T (min) in a branch is determined from the branch volume (V) and branch flow (Q) by:

$$T = \frac{V}{60Q} \quad (3.9)$$

The total amount of diffused radon added, C_a, may be obtained by integrating Eq. (3.7) directly.

Thus, $$C_a = \int_o^T \frac{J_s\ P}{A}\ dt$$

$$C_a = \frac{J_s\ P\ T}{A} \quad (\mathrm{Bq/m^3}) \quad (3.10)$$

If the different radon sources of an excavation are now considered, the total emanation rate, E_s, will be given by:

$$E_s = \sum_{i=1}^{n} J_i S_i \quad (\mathrm{Bq/s}) \quad (3.11)$$

where

J_i = emanation rate of source i
S_i = surface area of source i
n = number of different sources.

Therefore, in effect,

$$J_s = \frac{E_s}{PX} \qquad (3.12)$$

where

X = excavation length
PX = surface area of excavation

Hence by substitution into Eq. (3.8), converting the emanation rate to units of Bq/min by multiplying by 60, and noting that V (excavation volume) = AX, we get

$$WL_2 = \frac{60\ K\ E_s\ (T^{(y+1)} - b)}{(y+1)\ \ V} \qquad (3.13)$$

Equation (3.10) may be similarly modified to

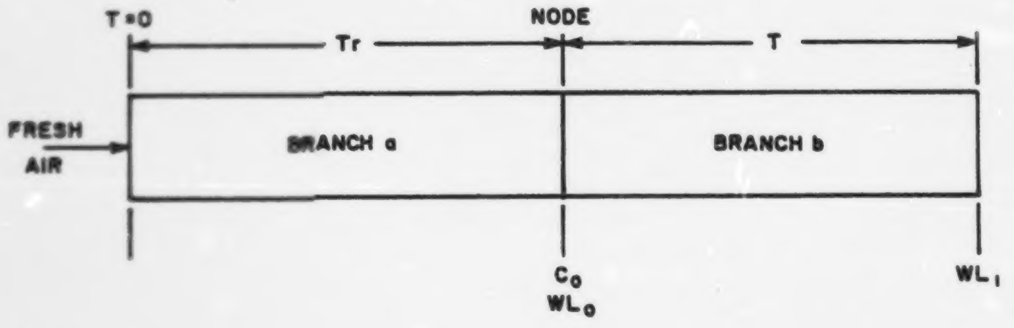
$$C_a = \frac{60\ E_s\ T}{V} \quad (\mathrm{Bq/m^3}) \qquad (3.14)$$

3.3 Age of Air Concept

Using the above formulae, we are now able to predict the radon and radon daughter levels at the end of an airway from radon emanation in that particular airway. Consider the simple case of two connected branches as shown schematically in Figure 3.4. The radon (C_a) and radon daughter (WL_2) concentrations from the first branch have been re-labeled C_o and WL_o at the node. The

FIGURE 3·4

TWO INTERCONNECTED BRANCHES



T = 0
NODE
Tr
T
FRESH AIR
BRANCH a
BRANCH b
C_0
WL_0
WL_1

equilibrium ratio E_r between radon and its daughters at the node is:

$$E_r = \frac{WL_o}{C_o} \times 3700 \qquad (3.15)$$

Due to the long half-life of radon (3.82 days) relative to the residence time of air in a branch (generally ten's of minutes or less), it can be assumed that the initial radon concentration, C_o, does not decrease as the air flows through the branch. Thus, the radioactive age of the air at the node (T_r) can be calculated for by combining equations (3.6) and (3.15).

$$E_r = \frac{WL_o}{C_o} \times 3700 = 3700\ K\ T_r^{\,Y}$$

$$T_r = \sqrt[Y]{\frac{E_r}{3700\ K}} \qquad (3.16)$$

Since equation 3.6 is fitted over three time ranges (Table 3.2) with three associated equilibrium ratio ranges, the known equilibrium ratio at the node is used to select the appropriate regression parameters.

The total radioactive age of the air (T_{tr}) at the end of the second branch can be approximated by adding the radioactive age of the air at the end of the first branch and the travel time in the second branch (T) or:

$$T_{tr} = T_r + T \qquad (3.17)$$

The total radioactive age T_{tr} can then be used to select the appropriate formulation of equation (3.6) to solve for WL_1:

$$WL_1 = KC_o\ (T_{tr})^Y \qquad (3.18)$$

Thus, by combining Eqs. (3.13), and (3.18), the total radon daughter concentration, WL_T, at a particular end point, will be given by:

$$WL_T = WL_1 + WL_2 \qquad (3.19)$$

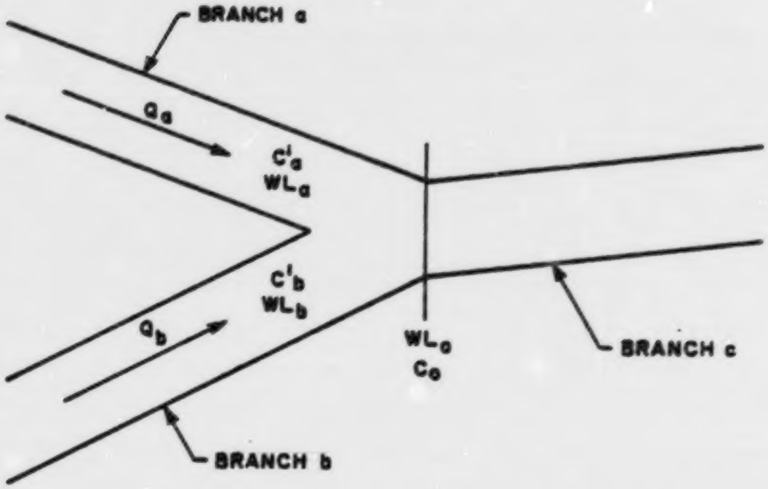
As previously noted, these relationships are based on the assumption that the activity of radon does not decrease during the period considered, an overestimation of less than 2%. It has also been assumed that the concentration of radon daughters is solely dependent on the laws of radioactive decay. The effects of removal mechanisms other than radioactive decay are discussed in Section 3.6.

Radon and radon daughter concentrations will change as air progresses through a mine. Given the initial radon and radon daughter concentrations at the beginning of a branch, the final concentration of radon and radon daughter concentrations due to this component can be estimated using the above formulations Initial radon and radon daughter concentrations at the junction of two or more branches can be obtained by volumetrically weighting the concentration from each branch. For example, consider the junction in Figure 3.5. The initial radon value C_o at the beginning of branch c is the flow weighted sum of the radon concentrations C_a' and C_b' at the end of branches a and b respectively or:

$$C_o = \frac{Q_a}{Q_a + Q_b} C_a' + \frac{Q_b}{Q_a + Q_b} C_b' \qquad (3.19)$$

FIGURE 3-5

JUNCTION OF BRANCHES


BRANCH a
Q_a
C'_a
WL_a
C'_b
WL_b
Q_b
WL_o
C_o
BRANCH c
BRANCH b

Similarly, the initial working level concentration for branch C (WL_o) is:

$$WL_o = \frac{Q_a}{Q_a + Q_b} WL_a + \frac{Q_b}{Q_a + Q_b} WL_b \qquad (3.20)$$

3.4 A Worked Example

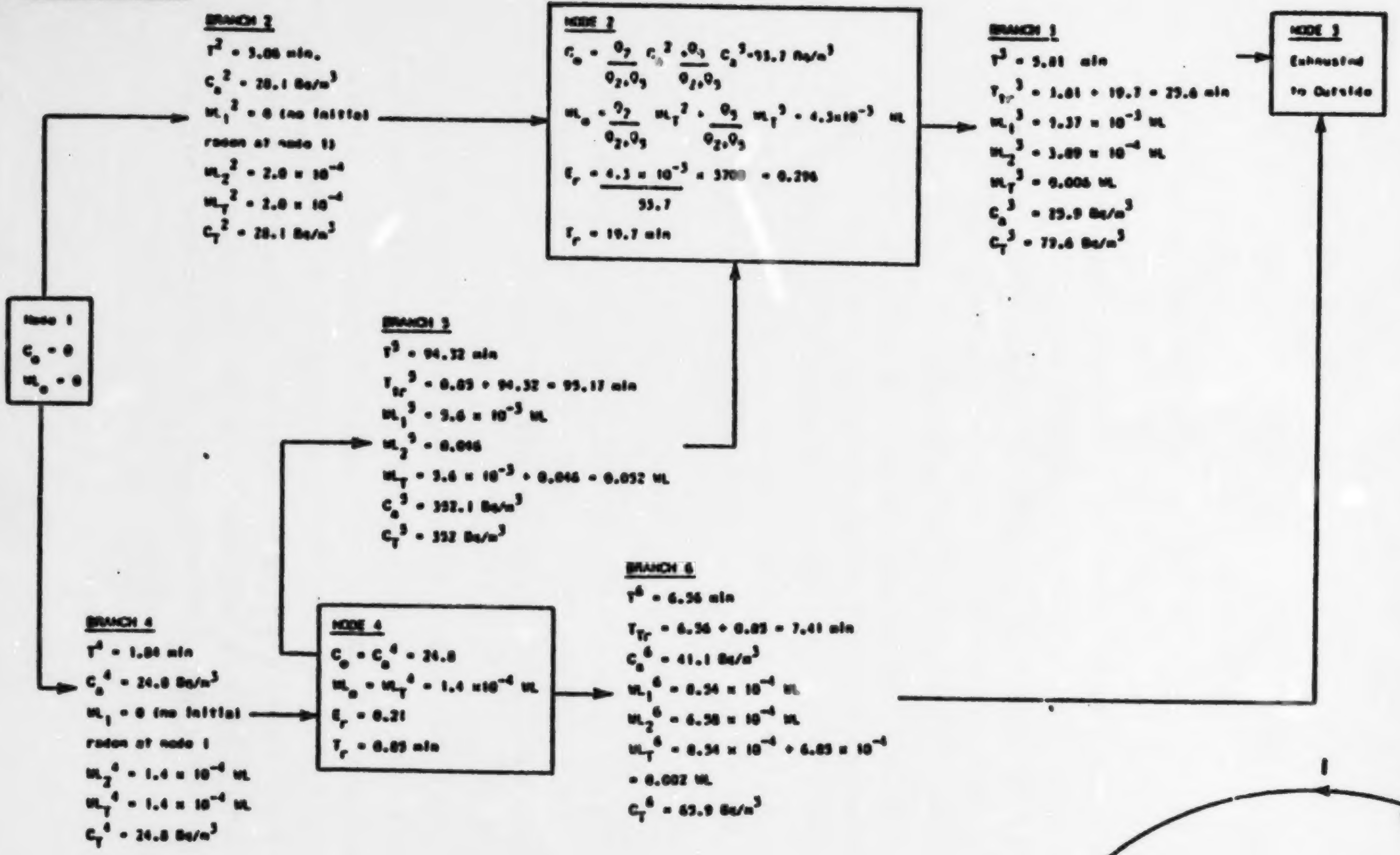
The same network used in the ventilation program example (Section 2.4) is used in this example. (This example is also discussed in the user's manuals for both the ventilation and radiation programs given in Sections 4 and 5 respectively). A network diagram and details of the worked example are shown in Figure 3.6. The calculations required to compute radon daughter levels can be subdivided into two classes: 1) preliminary calculations that are not dependent on the progressive transfer of radon and radon daughters from branch to branch, that is, branch volume, branch travel time and total branch radon emanation (Table 3.3); and 2) calculations that are dependent on the progressive transfer of radon and radon daughter concentrations through the network.

This group of calculations can be further subdivided into node and branch calculations (Figure 3.6). Node calculations combine the radon and radon daughters from branches coming into that particular node and compute "initial" radon and radon daughter levels. These initial radon and radon daughter levels are then used to compute the radioactive age of the air based on the relative concentration of radon and radon daughters. These parameters are then passed on to the branch or branches immediately downstream from that node. Branch calculations use the radon level and the radioactive age calculated at the upstream node to predict radon daughter levels due to the initial

FIGURE 3-8

THE WORKED EXAMPLE

Network Calculations



Node 1
$C_o = 0$
$WL_o = 0$

BRANCH 2
$T^2 = 3.06$ min.
$C_a^2 = 28.1$ Bq/m^3
$WL_1^2 = 0$ (no initial radon at node 1)
$WL_2^2 = 2.0 \times 10^{-4}$
$WL_T^2 = 2.0 \times 10^{-4}$
$C_T^2 = 28.1$ Bq/m^3

NODE 2
$C_o = \frac{Q_2}{Q_2+Q_5} C_a^2 + \frac{Q_5}{Q_2+Q_5} C_a^5 = 53.7$ Bq/m^3
$WL_o = \frac{Q_2}{Q_2+Q_5} WL_T^2 + \frac{Q_5}{Q_2+Q_5} WL_T^5 = 4.3 \times 10^{-3}$ WL
$E_r = \frac{4.3 \times 10^{-3} \times 3700}{53.7} = 0.296$
$T_r = 19.7$ min

BRANCH 3
$T^3 = 3.81$ min
$T_{Tr}^3 = 3.81 + 19.7 = 23.6$ min
$WL_1^3 = 5.37 \times 10^{-3}$ WL
$WL_2^3 = 3.89 \times 10^{-4}$ WL
$WL_T^3 = 0.006$ WL
$C_a^3 = 25.9$ Bq/m^3
$C_T^3 = 79.6$ Bq/m^3

NODE 3
Exhausted to Outside

BRANCH 5
$T^5 = 94.32$ min
$T_{Tr}^5 = 0.85 + 94.32 = 95.17$ min
$WL_1^5 = 5.6 \times 10^{-3}$ WL
$WL_2^5 = 0.046$
$WL_T = 5.6 \times 10^{-3} + 0.046 = 0.052$ WL
$C_a^5 = 332.1$ Bq/m^3
$C_T^5 = 332$ Bq/m^3

BRANCH 4
$T^4 = 1.04$ min
$C_a^4 = 24.8$ Bq/m^3
$WL_1 = 0$ (no initial radon at node 1
$WL_2^4 = 1.4 \times 10^{-4}$ WL
$WL_T^4 = 1.4 \times 10^{-4}$ WL
$C_T^4 = 24.8$ Bq/m^3

NODE 4
$C_o = C_a^4 = 24.8$
$WL_o = WL_T^4 = 1.4 \times 10^{-4}$ WL
$E_r = 0.21$
$T_r = 0.85$ min

BRANCH 6
$T^6 = 6.56$ min
$T_{Tr} = 6.56 + 0.85 = 7.41$ min
$C_a^6 = 41.1$ Bq/m^3
$WL_1^6 = 8.34 \times 10^{-4}$ WL
$WL_2^6 = 6.58 \times 10^{-4}$ WL
$WL_T^6 = 8.34 \times 10^{-4} + 6.83 \times 10^{-4}$
$= 0.002$ WL
$C_T^6 = 65.9$ Bq/m^3

BRANCH PARAMETERS

C_a^n = radon concentration due to emanation in nth branch (eqn. 3.14)
WL_1^n = radon daughter concentration in nth branch due to radon emanation in previous branches (eqn. 3.10)
WL_2^n = radon daughter concentration in nth branch due to radon emanation in nth branch (eqn. 3.13)
WL_T^n = total working levels in nth branch = $WL_1^n + WL_2^n$ (eqn. 3.11)
T^n = travel time in nth branch (eqn. 3.9)
T_{Tr}^n = total radioactive age in nth branch (eqn 3.17)
C_T^n = radon concentration at end of nth branch

NODE PARAMETERS

WL_o = initial radon daughter concentration of node
C_o = initial radon concentration of node
E_r = equilibrium ratio (eqn. 3.15)
T_r = radioactive age of air (eqn. 3.16)

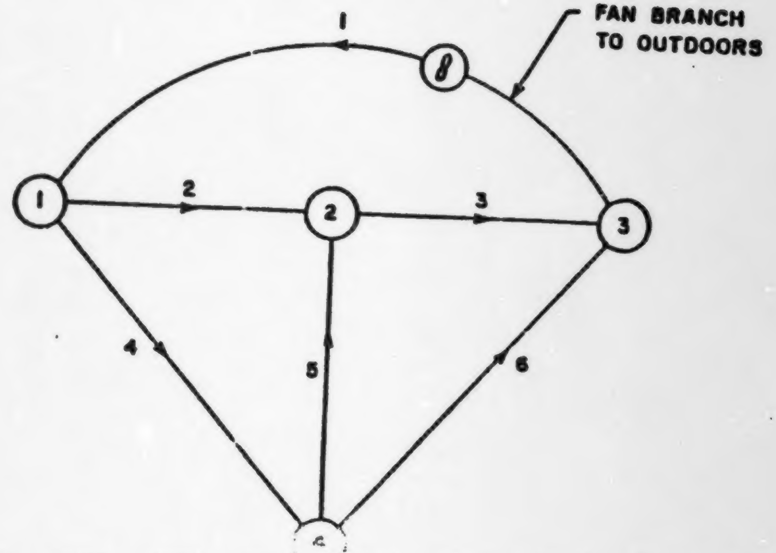


Table 3.3

Branch Parameters and Preliminary Calculations for Worked Example

Branch No.	Initial Node	Final Node	Flow (Q) (m^3)	Length (L) (m)	Cross Sect. (A) Area (m^2)	Volume (V) (m^3)	Travel Time (min)	Emanation Element Area(s) (m^2)	Emanation Rate ($Bq/m^2.s$)	Total Emanation (E_s) (Bq/s)
1	3	1	46.94	748	7	5236	1.86	-	-	-
2	1	2	21.98	577	7	4039	3.06	500 400 400	0.46 0.25 0.72	618
3	2	3	23.87	189	7	8323	5.81	500 400 400	0.46 0.25 0.72	618
4	1	4	24.96	393	7	2751	1.84	500 400 400	0.46 0.25 0.72	618
5	4	2	1.89	1528	7	10696	94.32	500 400 400	0.46 0.25 0.72	618
6	4	3	23.07	1299	7	9093	6.56	800 600 600	0.46 0.25 0.72	950
						Equations used:	(3.9)			(3.11)

radon concentration. The radon emanation within the branch is used to calculate the incremental radon and radon daughter levels within the branch due to this component. All of these data are then passed on to the downstream node where a new set of node calculations are performed. Although the procedure looks complicated the reader will find that, by starting at node 1 and following several branches, the process, detailed in Figure 3.6, is actually fairly straightforward.

3.5 Recirculation Loops

The procedure detailed in the worked example will solve a straightforward circuit in a single pass. However, special techniques are needed to handle recirculation loops, of which two types may be defined, 'simple' and 'compound'.

A simple recirculation loop consists of a single closed path with the air flowing in the same direction throughout the loop (see Figure 3.7a). A compound loop exists if any branch is common to two or more simple recirculation loops (see Figure 3.7b). In practical terms, such conditions can only exist in paths containing fans or sources of natural ventilation pressure. In the ventilation of uranium mines, recirculation is to be avoided since the longer residence time of the recirculating air leads to increased growth of radon daughters. Nevertheless, such situations can and do occur and the program has the capability of determining the effects of recirculation.

If a loop is encountered, the non-recirculation loop section of the program will cycle indefinitely in its attempt to find a junction with all entering branches evaluated. To prevent this, a check-list is maintained of the nodes visited in the search since the last successful evaluation. If a node already on the check-list is encountered again, a recirculation loop has been

FIGURE 3·7a

SIMPLE RECIRCULATION LOOP

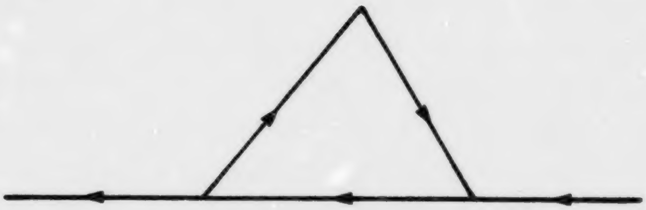
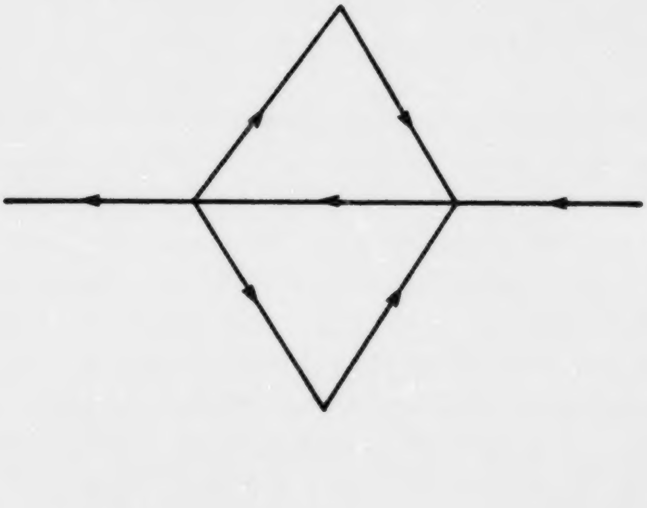


FIGURE 3·7b

COMPOUND RECIRCULATION LOOP



identified.

The newly found loop is traversed once more to check for other unevaluated entering branches. If none are found, the loop is a simple one and may be evaluated by starting with initial values of zero at one of the loop branches and iterating around the loop until the final radon concentration at the end of a selected test branch reaches a steady state. During this process, the parameters of the branches entering the loop at any junction are known and remain constant. These parameters are weighted together with the corresponding loop branch parameters which increase with each iteration. Branches leaving the loop are not calculated until steady state has been reached. After a simple recirculation loop has been evaluated in this manner, normal junction processing can continue.

If a second recirculation path is found during the check traverse of the first one, a compound loop condition is indicated. This requires a completely different approach. All the remaining unevaluated airways in the system are calculated in turn using all entering airways at each junction whether previously evaluated or not. This same sub-set of airways is then repeatedly evaluated until the radon concentration at the end of each airway has reached a steady state.

Programmatically, the algorithm for dealing with compound loops, once they have been found, is extremely simple. This technique could, of course, be used for the whole network. Tests on a network with no loops have shown that both methods give the same results. The node search algorithm, however, is much more effficient, particularly as the network size increases. In addition the search method identifies recirculation loops and prints a message indicating the node involved. Once the program goes into the compound loop solution mode, however, individual loops can no longer be identified. Since recirculation is the exception rather than the rule, most of the processing is usually done with the efficient algorithm.

Figure 3.8 illustrates an interesting phenomenon that can occur in a recirculation loop. As the flow in branches 25 and 26 decreases, the travel times and radon daughter levels increase in those branches. This "aged" air is returned to the main passageway, branch 23. The smaller the flow in branches 25 and 26, the higher the radon daughter concentration in branch 23. However, when the flow in branches 25 and 26 reaches zero, the radon daughter level in branch 23 quickly drops to the level due to emanation in that particular branch alone. Such a recirculation branch with a low flow can be used to simulate a leaking bulkhead or a unused section of a mine that has a very weak driving force that induces recirculation. It should also be noted that locally high concentrations can occur at node 51 before full mixing takes place. This could cause potential high exposures to workers in the area. Since the model assumes instantaneous mixing at the node, this effect will not be apparent from the model output.

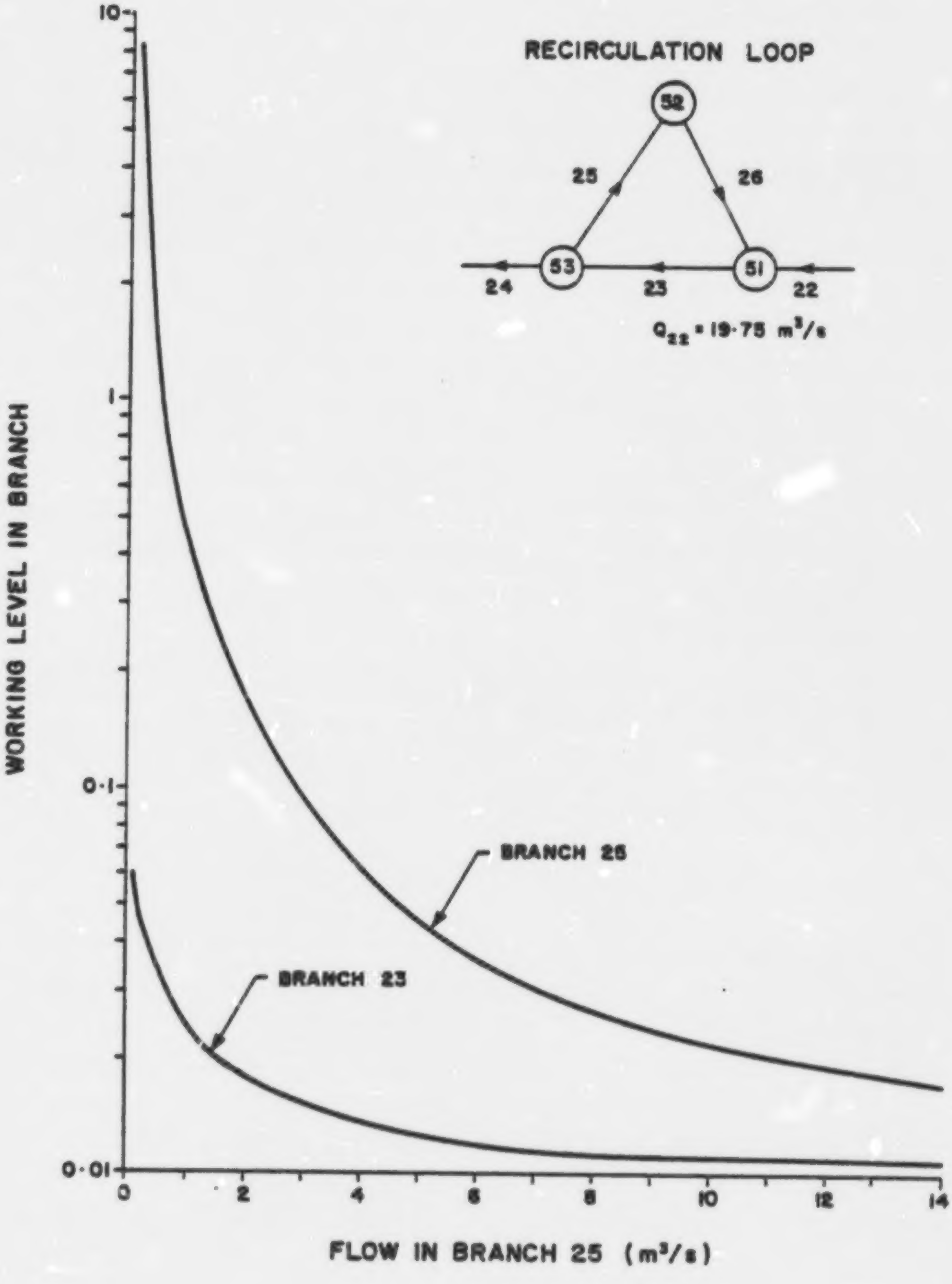
3.6 Limitations of Approach

3.6.1 Neglect of Physical Removal Mechanisms

The equations used in this study to model the concentrations of radon daughters in mine atmospheres are derived directly from the laws of radioactive decay and ventilation dynamics. It is assumed that no other removal mechanisms which could influence the concentrations are operable. In a real mine however, additional variables such as the plateout of unattached (to aerosols) daughters to mine surfaces and the deposition of attached daughters also influence the ultimate airborne

FIGURE 3·8

WORKING LEVEL - FLOW RELATIONSHIP IN A RECIRCULATION LOOP



concentration. While various studies and experimental measurements have been conducted to characterize these effects, the number of parameters involved and the data required preclude the establishment of exact quantitative relationships for modelling purposes, at least for operating underground mines. It is worthwhile, however to estimate the magnitude of this effect.

The effective deposition rate (or removal by plateout) of unattached (free) radon daughters is a nominal 0.1 to 1.0 cm/s for typical mine atmospheres (e.g. Jacobi 1972; Porstendorfer et al 1978). A substantial fraction of the unattached radon daughters may be removed from the mine atmosphere by this mechanism. However since the unattached fraction generally averages 5% or less (ICRP 1977; Jacobi and Eisfeld 1981) (depending on the aerosol concentration, humidity, ventilation rate, etc.), the unattached fraction has little influence on the overall concentration of radon daughters. (This would not apply to house atmospheres which generally have larger unattached fractions due to lower particulate levels).

The deposition velocity of the attached fraction has variously been reported as 10^{-2} to 10^{-3} cm/s (e.g. Porstendorfer et al 1978), or at least an order of magnitude less than for the unattached fraction, and the resultant effect on the concentration of attached daughters is correspondingly reduced. The deposition rate is dependent on factors such as deposition velocity, surface-to-volume ratio, and ventilation rate but is not likely to exceed a value of 0.01/min for most mining conditions (Jacobi 1972). This rate can be applied to the normal rates of radioactive decay for the radon daughters and the buildup of radon daughters from the radon can be recalculated. The results are shown on Figure 3.9 for various values of the deposition removal rate.

FIGURE 3-9

EFFECT OF DEPOSITION ON RADON DAUGHTER INGROWTH

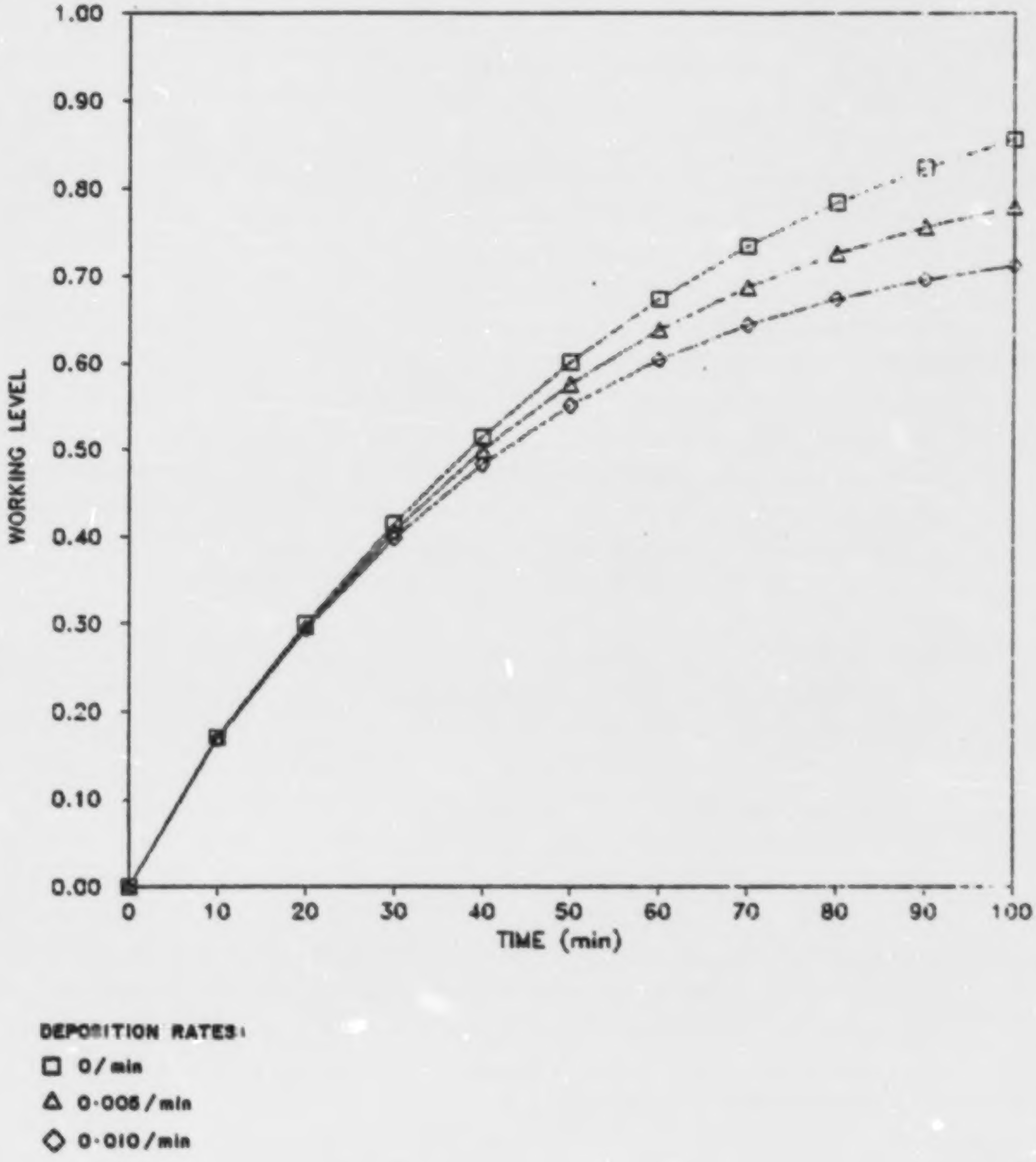


Figure 3.9 demonstrates that for normal residence times (<40 min) the effect of this removal mechanism is small (<10%). The effect is also small relative to the degree of accuracy of the ventilation and radiation prediction models.

3.6.2 Approach to Mixing Air Flows

As detailed in Section 3.3, the concentrations of radon and radon daughters at the junction of two or more branches are calculated by volumetrically weighting the concentration at the end of each branch. Other than the problem of initial mixing, this is an exact calculation for the resulting radon concentration but an approximation for the radon daughter concentration, as discussed below.

The approach taken in this study is to use the WL unit for radon daughter concentrations. In actuality, the WL value results from the concentrations of individual radon daughters. The resulting concentrations of daughters at a junction are thus the volumetrically weighted values of the individual daughter concentrations entering the junction, not the volumetrically weighted WL values. A given WL value does not uniquely determine individual radon daughter concentrations. However, the error introduced by the WL-weighting approach is likely to be small relative to the other input "errors" in the model (e.g. emanation rates).

4.0 USER'S MANUAL FOR THE VENTILATION PROGRAM VENTPROG

4.1 Program Abstract

The mine ventilation computer program (VENTPROG) calculates flows, pressure drops and power requirements for a mining ventilation network. Input information on each branch in the network is required, that is; branch node numbers, branch type, branch resistance or branch geometry and friction factor, preliminary flows and fan curves. These data are specified by the user in two disk files. The program sets up a series of loops or meshes and using an iterative solution called the Hardy Cross method, solves the circuit. Output from VENTPROG can be routed to a data set to provide input for the radon level program, VENTRAD. Control of the ventilation program is interactive, with the user operating the program through a simple series of control codes entered at the terminal. The following user's manual first details the ventilation program's input requirements then the output interpretation method. Two worked examples are also provided.

4.2 Model Inputs

Inputs to VENTPROG take two forms, either data manually entered at the terminal or disk files. The terminal inputs control the functions the user desires the program to perform. Two separate input disk files contain branch and fan data. Descriptions of the content and format of the various inputs follow.

4.2.1 Terminal Inputs

VENTPROG is composed of a series of subroutines that are called up by control codes entered at the terminal. The control codes execute the commands shown in Table 4.1

Table 4.1

Control Codes Entered at the Terminal

Control Code	Command
0	Print Menu of Control Codes at Terminal
1	Read Branch File
2	Solve Circuit
3	Read Fan Data
4	Print Mesh Data
5	Create New Branch File (this file can be used as input to the radon level program)
6	Suppress Listing of Branch File Input Data
7	End Program

The user would typically enter control code 1 to read the branch data. Then, if there are fans in the circuit, control code 3 would be entered to read the fan data. The user would then enter control code 2 to solve the circuit. If the user needs an output data set to use as input to the radon level program, control code 5 would then be entered.

4.2.2 Branch Inputs (File 5)

The subroutine LCTDT, activated by control code 1, reads the branch data, that is the physical characteristics of the branches, from file 5. A general description of the parameters in the branch file and their format is shown in Table 4.2. A more detailed description of how to set up a network follows.

In order to set up a branch input file the user must first set up a mine ventilation network, that is, a mathematical representation of a real mine ventilation system. A mine ventilation network is essentially a collection of airways or branches and data associated with them. Each branch has two endpoints , an initial and a final node. Each node is assigned a unique integer value. If the direction of flow in the branch goes from the initial to the final node the flow is positive and vice versa. The program handles the five different branch types shown in Table 4.3.

Branch resistances can be calculated by the user and directly input in the branch file or, alternatively, if the value -1.0 is input as a branch resistance the resistances are calculated by the program using the equation:

$$R = \frac{KP\ (L+L_e)}{A^3}$$

where

K = empirical friction factor (kg/m^3)
P = perimeter of branch (m)
L = length of branch (m)
L_e = equivalent length accounting for shock factors associated with branch (m)
A = area of duct (m^2)

(see Hartman (1982) for full explanation of resistance calculations)

These parameters must be input to the program if the user chooses to use the above resistance calculation method. This simplifies the calculation of the resistances as well as providing branch area and length data in an output file for input to the radon level program. Alternatively, if the user has already calculated the resistances, the branch lengths and areas must be passed on to the radon daughter level program, hence these parameters must also be specified as inputs.

The next step in setting up a branch file is to specify preliminary flows. The closer these preliminary flows are to the final flows, the fewer iterations are required in the Hardy Cross method.

If a branch has either a regulator, a fan or imposed flow additional information must be input. If the user has specified that a branch has a regulator the maximum allowable flow must be specified. If a branch is specified as having a fan, the fan number must be specified so that the correct fan curve is referenced by the program, in the fan curve data file discussed in the next section. Finally, if a branch has an imposed flow, the flow must be specified and must be the same as the preliminary flow estimate.

Table 4.2

Branch Data File Content and Format (File 5)

Column No.	Format	Parameter Description
Line 1 - Program Instruction:		
1-10	I10	Maximum no. of iterations allowed in Hardy Cross method (recommended value = 30)
11-20	F10.5	Required precision of solution in m^3/s (recommended value = 0.005 m^3/s)
21-30	I10	Maximum number of iterations allowed in recirculation loop section of radon program (value passed through the ventilation program)
31-40	I10	Number of detailed printouts of recirculation loop iterations allowed in radon program (value passed through ventilation program)
Line 2 - Branch Information:		
1-5	I5	Initial node of branch
6-10	I5	Final node

Table 4.2 (continued)

Column No.	Format	Parameter Description
11-20	I10	Branch type - numeric code 1 = normal, 2 = regulator, 3 = fan, 4 = imposed flow and 5 = linear characteristic
21-30	F10.4	Branch resistance (if resistance is to be calculated from branch geometry enter -1.0)
31-40	F10.4	Initial flow estimate (m^3/s)
41-50	F10.4	Additional data for regulator branches - maximum allowable flow. For fan branches - fan curve number. For imposed flow branches - imposed flow (equal to estimated flow)
51-60	F10.4	Natural Ventilation Pressure (pa) - generally set to zero
61-70	F10.4	Branch length (To be used if lengths are needed as an output file to be used in the radon level program and the resistance = -1.0 option was not specified).
71-80	F10.4	Branch cross sectional area (To be used if areas are needed as an output file to be used in the radon level program and the resistance = -1.0 option was not specified)

Table 4.2 (continued)

Column No.	Format	Parameter Description

Line 3 - Branch Geometry and Friction Factor (use only if resistance is set to -1.0 in previous line):

Column No.	Format	Parameter Description
1-10	F10.4	Branch length (m)
11-20	F10.4	Equivalent length (m)
21-30	F10.4	Branch area (m^2)
31-40	F10.4	Branch perimeter (m)
41-50	F10.4	Branch friction factor (kg/m^3)

Information for successive branches is input using the same formats as line 2 and the optional line 3. The end of the file is signified by a zero.

Table 4.3

The Five Branch Types

BRANCH NUMBER	BRANCH DESCRIPTION
Branch Type 1:	Normal branch, in which the pressure drop will be calculated as $P = R Q^2$
Branch Type 2:	Regulator branch $P = R Q^2$ with Q always less than or equal to the specified regulator flow.
Branch Type 3:	Fan branch. $P = R Q^2 + F(Q)_i$ $F(Q_i)$ = pressure of fan (i) at flow rate Q from fan characteristic curve. (see section 4.2.3)
Branch Type 4:	Imposed flow (fixed quantity) branch $Q = Q_1$. Use for initial run to size fans. Flow in the branch will be maintained at Q_1.
Branch Type 5:	Linear characteristic branch where $P = -R_L Q$ and R_L = linear resistance. Use for flow through caved ground or high resistance doors (Laminar flow).

There are certain circuits that are not solvable by the ventilation program. The program will automatically print an error message if it encounters one of these circuits. Essentially, the program cannot solve a circuit if there is more than one regulator or fixed flow branch in a closed loop and there are no branches of any other type that provide an alternative path for the flow. This concept will become clear by considering Figure 4.1. The flows around a closed loop must balance to equal zero. If the flows have been fixed to some value in two or more places in a loop with no alternative paths, the circuit cannot be solved.

4.2.3 Fan Characteristic Curve Inputs (File 7)

The subroutine CBVT activated by control code 3, reads the fan data from the fan data file (File 7). The general content and format is indicated in Table 4.4. The fan characteristic curves are described in the following section.

Fan Characteristic Curves

A fan is represented in the circuit by a plot of fan pressure vs. flow called a fan characteristic curve that is associated with a fan branch (see Figure 4.2). The fan branch also has a normal resistance and an attendant RQ^2 pressure drop. For a specified flow, the program interpolates a fan pressure and slope of the fan curve between the two fan data points bracketing the flow for use in the Hardy Cross method.

Fan characteristic curves are referenced by a signed real number with a positive sign indicating that the fan is blowing air from the initial to the final node and vice versa. Several fan characteristic curves may be successively associated with a

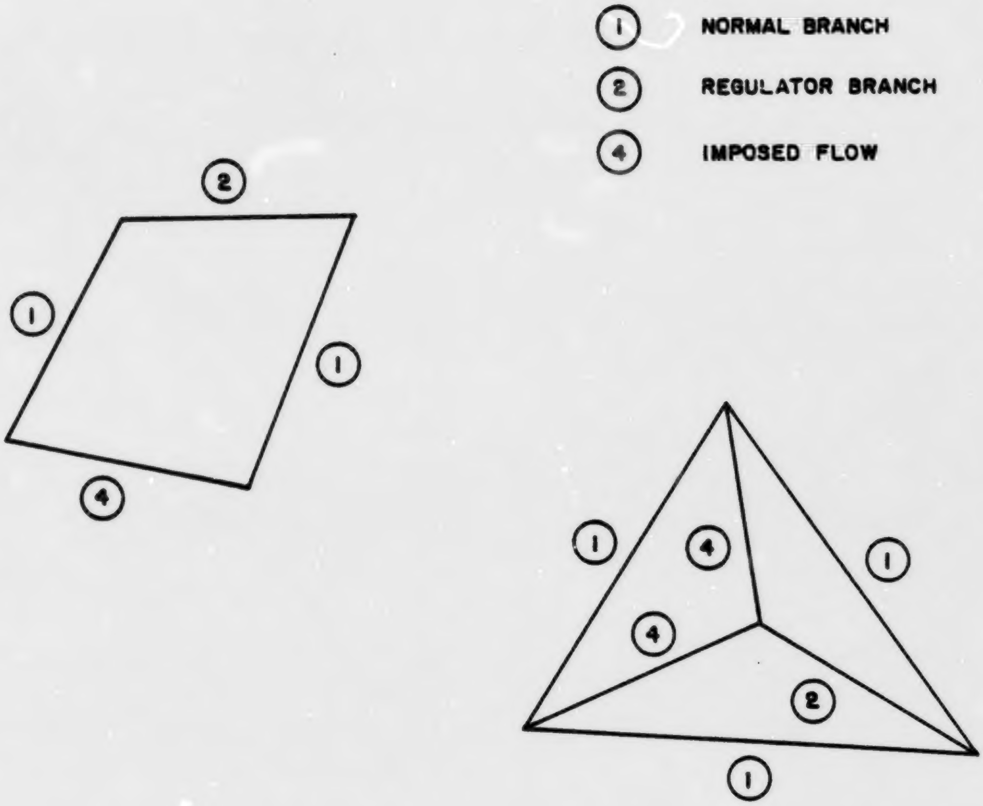


FIGURE 4·1

UNSOLVABLE CIRCUITS

① NORMAL BRANCH
② REGULATOR BRANCH
④ IMPOSED FLOW
②
①
①
④
①
④
①
④
②
①

Table 4.4

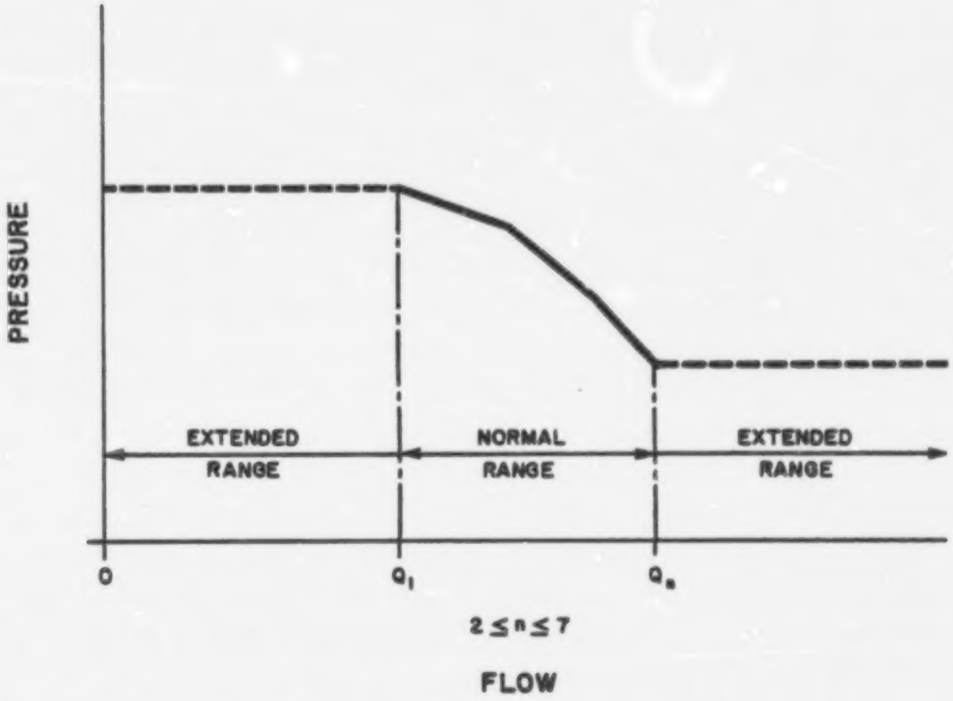
FAN CHARACTERISTIC CURVE DATA FILE CONTENT AND FORMAT

File 7- Fan File

Column	Format	Line 1 - Ascending Flow values
2-10	F9.0	- curve number
11-20	F10.4	- 2 to 7 flow values
21-30	F10.4	
etc.		
		Line 2 - Descending Pressure Values
11-20	F10.4	- 2 to 7 pressure values
21-30	F10.4	
etc.		
		Subsequent lines
		(as above, two per fan)
		Final line
1		Blank or zero

FIGURE 4·2

FAN CHARACTERISTIC CURVE



single branch, and the same fan characteristic may be simultaneously associated with a number of branches.

A fan characteristic curve is defined by a series of 2 to 7 points in the positive quadrant of the pressure vs. flow diagram such that:

$Q_i < Q_{i+1}$

and $P_i \geq P_{i+1}$

The first and last values of flow define the normal operating range of the fan.

In the program, the fan characteristic curve is extended beyond the limiting flow rates at constant pressure. If the solution is found on the extended range, a message is given.

4.3 Program Output

The following section describes the output from VENTPROG. A full listing of output is shown for the worked examples in the next section in Tables 4.8 and 4.11. It is suggested that the reader review them while reading this section. The first output information to appear will be the input branch data, the number of branches, branch types and nodes and the input fan data. The program will then print the meshes or loops selected if control code 4 is specified.

After the meshes are printed the following information is listed:

1. the number of iterations made
2. the sum of the absolute values of the last correction factors for all the meshes.

3. a table is then printed containing the information listed below:

col 1-3	branch and node numbering information
col. 4	the flow (with sign indicating direction)
col. 5	the pressure drop due to resistance i.e. R Q^2 (Pa)
col. 6	the pressure generated by a fan or lost through a regulator (PRESS REG/FAN) for branches of type (2), (3), or (4).
col. 7	the air power dissipated in the branch i.e. PQ (kW)
col. 8	the total pressure change in the branch (PRESS TOTAL) which is the sum of the pressure loss due to resistance, and natural ventilation pressure and any loss or gain due to regulator or fan. (If the pressure has the same sign as the branch flow, a pressure increase (due to a fan) is indicated).
col. 9	branch type
col. 10	resistance
col. 11	Natural ventilation pressure
col. 12	-for regulator, fan and imposed flow, branches, a message is printed -for regulator branches it is indicated whether the flow is restricted to the maximum allowed in that branch or not. -for fan branches the air power generated by the fan, and a message is printed to indicate if the fan is operating out of the normal range -for imposed flow branches it is indicated if the branch is operating as a fan (in which case the fan power is also given), or if it is operating as a regulator)

4. -the total power required by the circuit

5. -a convergence index which is a measure of the difference between the energy generated by the fans and

the energy consumed by the branches. (Would be equal to zero for perfect convergence)

The next table lists the branches with more than a 10% difference between the estimated and solved flows.

By specifying control code 5, an output file can be created (File 9) that can be used as an input file to the ventilation program or the radon daughter program.

4.4 Two Worked Examples

Two worked examples are examined in the following section. The first is a continuation of the highly simplified example presented in sections 2.4 and 2.7. The second is a more complex example that more closely approximates "real world" conditions. Input data sets for both of these examples are included on disk in order that the user may test the program.

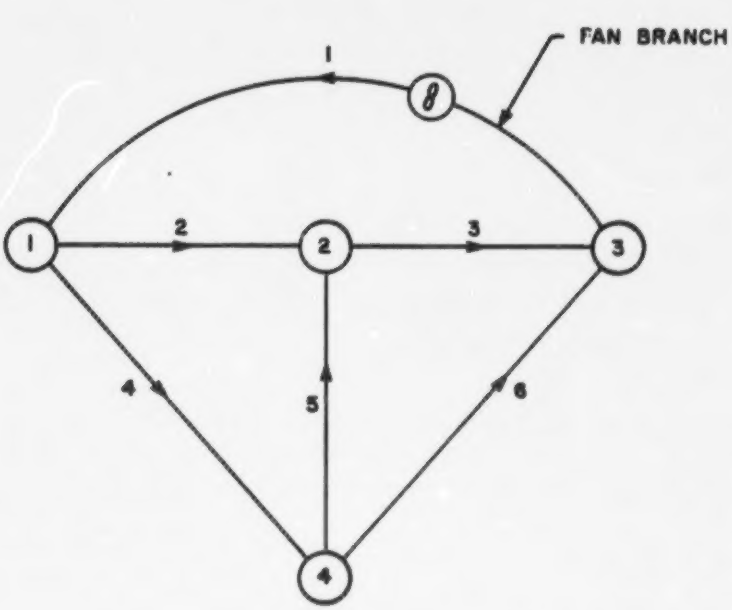
Example 1

Consider the network indicated in Figure 4.3. In this example the branch resistances are calculated by the program hence the branch geometry and friction factor are included. The branch input data file is shown in Table 4.5 utilizing the format and content indicated in Table 4.2.

The fan pressure in branch 1 is a constant 1990 Pa. Since the fan pressure is a constant, two flow values of 40 and 60 m^3/sec were arbitrarily chosen to correspond to the same fan pressure. This will yield a slope of zero for the fan curve. This zero slope will be used in the numerator in the Hardy Cross method by the program. These data were entered into a fan input data set using the format indicated in Table 4.4. The fan input data set is

FIGURE 4·3

NETWORK DIAGRAM FOR EXAMPLE I



FAN BRANCH
1
8
2
3
4
5
6
1
2
3
4

Table 4.5

BRANCH FILE FOR EXAMPLE 1

Maximum Number of Iterations

Required Precision

Maximum Number of Iterations Radon Program

Detailed Printout Code for Radon Program

```
          5          0.1     30              0
   3      1             3   -1.0            46.4        1.0        0.0
 748.         200.           7.0        9.4            .0223
   2      1             1   -1.0            21.7        0.0        0.0
 577.         200.           7.0        9.4            .0223
   2      3             1   -1.0            23.5        0.0        0.0
1189.         200.           7.0        9.4            .0223
   1      4             1   -1.0            24.7        0.0        0.0
 393.         200.           7.0        9.4            .0223
   4      2             1   -1.0             1.9        0.0        0.0
1528.         200.           7.0        9.4            .0223
   4      3             1   -1.0            22.8        0.0        0.0
1299.         200.           7.0        9.4            .0223
0
```

Natural Ventilation Pressure (pa)
Additional Information
Branch Friction Factor (kg/m^3)
Preliminary Flow Estimate (m^3/sec)
Branch Perimeter (m)
-1.0 indicates resistance to be calculated from branch geometry on next line
Branch Area (m^2)
Branch Type
Equivalent Length (m)
Final Node
Initial Node
Branch Length (m)
End of File

Table 4.6

FAN INPUT FILE FOR EXAMPLE 1

```
1.0      40.0      60.0
       1990.7    1990.7
0
```

Fan Flows on Line
Fan Pressure on Line
Fan Number
End of File

shown in Table 4.6.

After the input data files were set up and a run file appropriately referencing the input and output data sets to the program was created, the program was run. A printout of what occurred at the terminal, that is the order of entry of control codes is shown in Table 4.7. Note the use of control code five to create an output data file (File 9). This file can be used as input to the ventilation program if the user desires to make changes on an input to the radon daughter level program. Indeed the file will eventually be used in example 1 for the radon daughter program in the next chapter. A full output listing of the program is shown in Table 4.8.

Example 2

Example 2 is an analysis of a more complex network. A network diagram is shown in Figure 4.4. In this example resistances were directly input to the program with the branch length and cross sectional area input at the end of each line. These data were transferred through the program into the output file created by control code 5 and were not used in computations in VENTPROG. This file will eventually be used in the radon daughter level model Example 2. The branch and fan input files are shown in Table 4.9 and 4.10.

The terminal inputs were identical to the last session, hence a table of them is not included. The output is listed in Table 4.11.

Table 4.7

WHAT APPEARS ON THE SCREEN FOR EXAMPLE 1

```
 INITIAL DATA
 -----------

          MAX.NUMBER OF ITER.          5
          PRECISION REQUIRED       0.10000
          COEFFICIENT (P*Q TO )KW   0.00100
 ENTER CONTROL CODE:
 IF YOU WANT A MENU OF CODES,' TYPE "0"

End of file. Cominput. (Input from terminal.)
0
 ENTER CONTROL CODE:
 1 = READ CIRCUIT DATA
 2 = SOLVE CIRCUIT
 3 = READ FAN DATA
 4 = PRINT MESH DATA
 5 = CREATE NEW BRANCH FILE
 6 = SUPPRESS LISTING
 7 = TERMINATE JOB
1

 *******************CONTROL CODE # 1***************************

 CIRCUIT CHARACTERISTICS
 -----------------------

                        NODES              4
                        BRANCHES: TOTAL    6
                               FAN         1
                               REGULATED   0
                               FIXED       0
 ENTER CONTROL CODE:
 IF YOU WANT A MENU OF CODES,' TYPE "0"
3

 *******************CONTROL CODE # 3***************************

 FAN CHARACTERISTIC CURVES
 -------------------------


 FAN  CURVES IN :

          1.
 ENTER CONTROL CODE:
 IF YOU WANT A MENU OF CODES,' TYPE "0"
2

 *******************CONTROL CODE # 2***************************
 NUMBER OF ITERATIONS     3
 LAST SUM OF ERRORS    0.05625
 ENTER CONTROL CODE:
 IF YOU WANT A MENU OF CODES,' TYPE "0"
5

 *******************CONTROL CODE # 5***************************
         5   0.10000   0.00100
 ENTER CONTROL CODE:
 IF YOU WANT A MENU OF CODES,' TYPE "0"
7

 *******************CONTROL CODE # 7***************************

 **** STOP
```

Table 4.8

OUTPUT FILE FOR EXAMPLE 1

```
********************CONTROL CODE # 1*************************************************************
DATA READING
____________

BRANCH #  IN.  FIN.    TYPE    RESISTANCE         FLOW          ADD.DATA        NAT.PRESS.
   1       3     1      3       0.58100          46.400          1.000            0.000
   2       1     2      1       0.47500          21.700         -0.000            0.000
   3       2     3      1       0.84900          23.500          0.000            0.000
   4       1     4      1       0.36300          24.700          0.000            0.000
   5       4     2      1       1.05600           1.900          0.000            0.000
   6       4     3      1       0.91600          22.800          0.000            0.000
CIRCUIT CHARACTERISTICS
_______________________

                    NODES              4
                    BRANCHES: TOTAL    6
                              FAN      1
                              REGULATED 0
                              FIXED    0
********************CONTROL CODE # 3*************************************************************

FAN CHARACTERISTIC CURVES
_________________________

               CURVE #   1.     FLOW        PRESSURE

                              40.000        1990.700

                              60.000        1990.700

FAN  CURVES IN :

        1.
********************CONTROL CODE # 4*************************************************************

KIRCHOFF EQ. LOOPS
__________________

 LOOP #     BRANCHES
0  1           1
                  4    5    3
0  2           6
                 -3   -5
0  3           2
                 -5   -4

 ********************CONTROL CODE # 2*************************************************************
 NUMBER OF ITERATIONS   2
 LAST SUM OF ERRORS   0.20876


 RESULTS
 _______
   BR. INI. FINAL  BRANCH      PRESS.       PRESS.       RES.KW      PRESS.    BRANCH   RES.       NVP.      ADD.
   NO. NODE NODE   FLOW        RESIS.       REG/FAN                  TOTAL     TYPE                          DATA
    1    3    1   46.869    -1276.284     1990.700      59.818     714.416     3     0.58100   0.00000    1.00
                                                                                                           FAN  93.30

    2    1    2   21.957     -228.996        0.000       5.028    -228.996     1     0.47500   0.00000    0.00
    3    2    3   23.832     -482.201        0.000      11.492    -482.201     1     0.84900   0.00000    0.00
    4    1    4   24.912     -225.285        0.000       5.612    -225.285     1     0.36300   0.00000    0.00
    5    4    2    1.875       -3.714        0.000       0.007      -3.714     1     1.05600   0.00000    0.00
    6    4    3   23.037     -486.124        0.000      11.199    -486.124     1     0.91600   0.00000    0.00
 TOTAL FAN POWER REQUIREMENT :       93.30 KW
  CONVERGENCE INDEX   15.6
```

FIGURE 4·4

NETWORK DIAGRAM FOR EXAMPLE 2

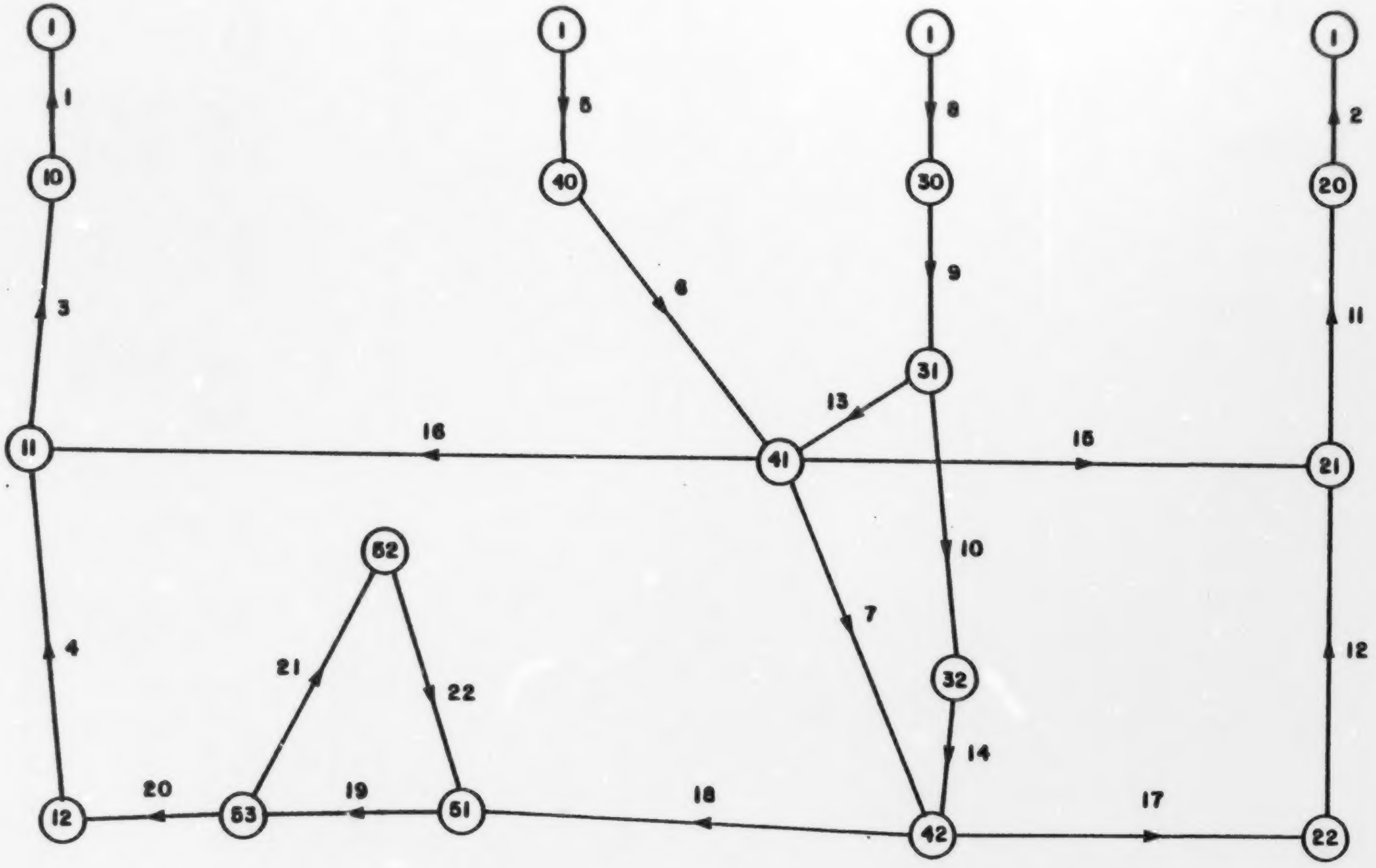

1
1
1
1
1
5
8
2
10
40
30
20
3
6
9
11
31
13
11
16
41
15
21
52
10
7
4
21
22
32
12
14
20
19
18
17
12
53
51
42
22

Table 4.9

BRANCH INPUT FILE FOR EXAMPLE 2

Maximum Number of Iterations	Required Precision	Maximum Number of Iterations Radon Prcgram	Detailed Printout Code for Radon Program
30	0.005	30	0

End of File	Initial Node	Final Node	Branch Type	Branch Resistance	Preliminary Flow Estimate (m³/s)	Additional Information	Natural Ventilation Pressure	Branch Length (m)	Branch Cross Sectional Area (m²)
	10	1	3	0.	50.0	1.0		0.	0.
	20	1	3	0.	50.0	1.0	0.0		
	10	11	1	0.013	-50.0			100.	10.
	11	12	1	0.013	-25.0			100.	10.
	1	40	1	0.0	50.0				
	40	41	1	0.017	50.0			300.	15.
	41	42	1	0.017	25.0			300.	15.
	1	30	1	0.0	50.0				
	30	31	1	0.003	50.0			100.	24.
	31	32	1	0.003	50.0			100.	24.
	20	21	1	0.013	50.0			100.	10.
	21	22	1	0.013	25.0			100.	10.
	31	41	1	0.005	25.0			50.	12.
	32	42	1	0.005	25.0			50.	12.
	41	21	1	0.050	25.0			500.	12.
	41	11	1	0.050	25.0			500.	12.
	42	22	1	0.050	25.0			500.	12.
	42	51	1	0.020	25.0			200.	12.
	51	53	1	0.010	50.0			100.	12.
	53	12	1	0.020	25.0			200.	12.
	53	52	1	0.010	25.0			100.	12.
	52	51	3	0.010	25.0	2.0		100.	12.
0									

Table 4.10

FAN INPUT FILE FOR EXAMPLE 2

```
1.     28.4    37.8    44.4    47.2    49.0    51.0    55.
       241.    179.    149.    125.    96.     74.     0.
2.     14.2    18.9    21.7    23.6    24.5    25.5    27.5
       110.    85.     70.     60.     45.     35.     0.
0.
```

Fan Flows on Line

Fan Pressure on Line

Fan Number

End of File

Table 4.11

OUTPUT FILE FOR EXAMPLE 2

```
********************CONTROL CODE # 1*********************************************************
DATA READING
------------

BRANCH #  IN.  FIN.   TYPE   RESISTANCE        FLOW          ADD.DATA       NAT.PRESS.
    1      10     1     3     0.00000        50.000          1.000          0.000
    2      20     1     3     0.00000        50.000          1.000          0.000
    3      10    11     1     0.01300       -50.000          0.000          0.000
    4      11    12     1     0.01300       -25.000          0.000          0.000
    5       1    40     1     0.00000        50.000          0.000          0.000
    6      40    41     1     0.01700        50.000          0.000          0.000
    7      41    42     1     0.01700        25.000          0.000          0.000
    8       1    30     1     0.00000        50.000          0.000          0.000
    9      30    31     1     0.00300        50.000          0.000          0.000
   10      31    32     1     0.00300        50.000          0.000          0.000
   11      20    21     1     0.01300        50.000          0.000          0.000
   12      21    22     1     0.01300        25.000          0.000          0.000
   13      31    41     1     0.00500        25.000          0.000          0.000
   14      32    42     1     0.00500        25.000          0.000          0.000
   15      41    21     1     0.05000        25.000          0.000          0.000
   16      41    11     1     0.05000        25.000          0.000          0.000
   17      42    22     1     0.05000        25.000          0.000          0.000
   18      42    51     1     0.02000        25.000          0.000          0.000
   19      51    53     1     0.01000        50.000          0.000          0.000
   20      53    12     1     0.02000        25.000          0.000          0.000
   21      53    52     1     0.01000        25.000          0.000          0.000
   22      52    51     3     0.01000        25.000          2.000          0.000

CIRCUIT CHARACTERISTICS
-----------------------

                NODES             16
                BRANCHES: TOTAL   22
                     FAN           3
                     REGULATED     0
                     FIXED         0

********************CONTROL CODE # 3*********************************************************

FAN CHARACTERISTIC CURVES
-------------------------

           CURVE #   1.      FLOW        PRESSURE

                            28.400       241.000

                            37.800       179.000

                            44.400       149.000
```

Table 4.11 (continued)

```
                                 47.200        125.000

                                 49.000         96.000

                                 51.000         74.000

                                 55.000          0.000
              CURVE #    2.        FLOW        PRESSURE

                                 14.200        110.000

                                 18.900         85.000

                                 21.700         70.000

                                 23.600         60.000

                                 24.500         45.000

                                 25.500         35.000

                                 27.500          0.000

FAN  CURVES  IN :

        1.
        2.

*******************CONTROL CODE # 2**************************************************************
NUMBER OF ITERATIONS     7
LAST SUM OF ERRORS    0.00744


RESULTS
-------
  BR.  INI.  FINAL  BRANCH      PRESS.      PRESS.       RES.KW      PRESS.   BRANCH   RES.       HYP.      ADD.
  NO.  NODE  NODE   FLOW        RESIS.      REG/FAN                  TOTAL    TYPE                          DATA
    1   10     1    49.211       0.000      93.684       0.000      93.684     3     0.00000    0.00000    1.00
*                                                                                                                FAN     4.6
    2   20     1    49.832       0.000      86.846       0.000      86.846     3     0.00000    0.00000    1.00
*                                                                                                                FAN     4.
    3   10    11   -49.211      31.482       0.000       1.549      31.482     1     0.01300    0.00000    0.00
    4   11    12   -19.754       5.073       0.000       0.100       5.073     1     0.01300    0.00000    0.00
    5    1    40    33.270       0.000       0.000       0.000       0.000     1     0.00000    0.00000    0.00
    6   40    41    33.270     -18.817       0.000       0.626     -18.817     1     0.01700    0.00000    0.00
    7   41    42    11.246      -2.150       0.000       0.024      -2.150     1     0.01700    0.00000    0.00
    8    1    30    65.773       0.000       0.000       0.000       0.000     1     0.00000    0.00000    0.00
    9   30    31    65.773     -12.978       0.000       0.854     -12.978     1     0.00300    0.00000    0.00
   10   31    32    31.601      -2.996       0.000       0.095      -2.996     1     0.00300    0.00000    0.00
   11   20    21   -49.832      32.282       0.000       1.609      32.282     1     0.01300    0.00000    0.00
   12   21    22   -23.093       6.933       0.000       0.160       6.933     1     0.01300    0.00000    0.00
   13   31    41    34.172      -5.839       0.000       0.200      -5.839     1     0.00500    0.00000    0.00
```

Table 4.11 (continued)

```
    14   32   42   31.601      -4.993       0.000       0.158      -4.993     1      0.00500     0.00000   0.00
    15   41   21   26.739     -35.750       0.000       0.956     -35.750     1      0.05000     0.00000   0.00
    16   41   11   29.456     -43.384       0.000       1.278     -43.384     1      0.05000     0.00000   0.00
    17   42   22   23.093     -26.664       0.000       0.616     -26.664     1      0.05000     0.00000   0.00
    18   42   51   19.754      -7.805       0.000       0.154      -7.805     1      0.02000     0.00000   0.00
    19   51   53   45.332     -20.550       0.000       0.932     -20.550     1      0.01000     0.00000   0.00
    20   53   12   19.754      -7.805       0.000       0.154      -7.805     1      0.02000     0.00000   0.00
    21   53   52   25.578      -6.542       0.000       0.167      -6.542     1      0.01000     0.00000   0.00
    22   52   51   25.578      -6.542      33.635       0.167      27.093     3      0.01000     0.00000   2.00
+                                                                                                           FAN    0.86
0TOTAL FAN POWER REQUIREMENT :      9.80 KW
 CONVERGENCE INDEX    0.0
1

 BRANCHES WITH MORE THAN 10$ FLOW CHANGE

   BR. INI. FINAL   NEW         OLD        CHANGE
   NO. NODE NODE    FLOW        FLOW
     4   11   12  -19.754     -25.000       5.246
     5    1   40   33.270      50.000     -16.730
     6   40   41   33.270      50.000     -16.730
     7   41   42   11.246      25.000     -13.754
     8    1   30   65.773      50.000      15.773
     9   30   31   65.773      50.000      15.773
    10   31   32   31.601      50.000     -18.399
    11   20   21  -49.832      50.000     -99.832
    12   21   22  -23.093      25.000     -48.093
    13   31   41   34.172      25.000       9.172
    14   32   42   31.601      25.000       6.601
    16   41   11   29.456      25.000       4.456
    18   42   51   19.754      25.000      -5.246
    20   53   12   19.754      25.000      -5.246
1

 NODE PRESSURES RELATIVE TO ATMOSPHERIC

     NODE  PRESSURE     NODE  PRESSURE     NODE  PRESSURE     NODE  PRESSURE     NODE  PRESSURE

        1     0.000       10   -93.684       11  -62.202       12   -57.129       20   -86.846
       21   -54.563       22   -47.631       30    0.000       31   -12.978       32   -15.974
       40     0.000       41   -18.817       42  -20.967       51   -28.771       52   -55.864
       53   -49.321

 *********************CONTROL CODE # 5*****************************************************************

 *********************CONTROL CODE # 7*****************************************************************
```

5.0 USER'S MANUAL FOR THE RADON DAUGHTER LEVEL PROGRAM VENTRAD

5.1 Program Abstract

A computer program has been written according to the principles outlined in Section 3.0, that simulates the growth of radon daughter concentrations through a mine airway network. The program, VENTRAD, is written in FORTRAN 77 and should be used in conjunction with the mine ventilation network analysis program VENTPROG which establishes the airflow distribution.

Two sets of input data are required by the program. First, the beginning and end nodes must be specified for each branch of the network, along with its length, cross-sectional area and airflow rate. Since the output from a ventilation network analysis program may have negative flow values, the input structure for VENTRAD tests for this condition and reverses the node order and airflow sign where a negative flow is encountered.

The second set of input data concerns airways with initially known concentrations of radon and radon daughter products and branch emanation characteristics. Known concentrations are specified together with their respective nodal locations. If a full mine circuit is being simulated, the contaminants in the intake air would be set to zero in this manner. For a sub-set of airways, intitial non-zero conditions may be specified. Filtration units at a node may also be specified.

In terms of emanation characteristics, pairs of values of the emanating surface area and the emanation rate are specified for each branch. This allows for the different components of radon entry into any one airway. For instance, in a seam-type deposit, the walls of a drift may be in ore while the roof and floor may be in waste. Additionally, there may be a layer of crushed ore on the floor which would have a higher emanation rate than a solid rock surface. This feature is also used for modelling emanations from water streams or muck-piles. Up to five emanation rates may be specified for each airway.

Ultimately the output of the program will provide the user with a listing of the input data as well as working levels, radon concentration, equilibrium ratios and radioactive ages for each branch.

5.2 Model Inputs

Two input files are required for VENTRAD, a branch file and an emanation characteristics file.

5.2.1 Branch File (File 7)

Since a balanced network is required by VENTRAD, it is recommended that the output file created by control code 5 of VENTPROG be used as the input branch file to VENTRAD. The branch file is file number 7 in the radon program. The format of this file is fully compatible with VENTRAD. The format for the branch file is indicated in Table 5.1.

5.2.2 Branch Emanation Characteristics (File 8)

The branch emanation characteristics input file (file number 8 in the radon program) allows the user to input information for nodes with known radon and radon daughter levels or filtration units. The user also can specify up to five emanation rates and areas for each branch. The format of the branch emanation characteristics file is illustrated in Table 5.2.

Table 5.1

Format and Content of Branch File for VENTRAD (File 7)

Column	Format	Parameter Description
Line 1 - Program Instruction:		
21-30	I10	Number of iterations
31-40	I10	Number of printouts of recirculation loop iterations (If parameter is greater than number of iterations, then all iterations will be printed out)
Line 2:		
1-5	I5	Initial node
6-10	I5	Final node
31-40	F10.5	Balanced flowrate (m^3/s)
61-70	F10.5	Branch length (m)
71-80	F10.5	Branch cross sectional area (m^2)

Successive Line 2's are entered for all branches.

Table 5.2

Format and Content of Emanation Characteristics File For VENTRAD (File 8)

Column	Format	
Line 1 - nodes with known input parameters:		
1-10	I10	Node number with known input parameter, either radon and radon daughter levels or a filtration unit. This parameter can be set to 1 in the 1st line to initialize all radon and radon daughter levels in surface air intake branches to zero.
11-20	F10.3	Known radon level. If a negative value is input, the program assumes a filtration unit at the node
21-30	F10.3	Known radon daughter level. If previous parameter was negative, the efficiency of the filtration unit is entered here.

Successive lines as above are input for each node with known radon and radon daughter levels, or filtration units.

Line 2:

A blank line or zero signifies the end of the above section of the input file

Table 5.2 (continued)

Column	Format	
Line 3:		
1-6	I6	Beginning node of branch
9-14	I6	End node of branch
Line 4:		
1-4	I4	Number of different emanating components
Line 5: emanation characteristics		
1-8	F8.1	Area of emanating component (m^2)
9-16	F8.3	Emanation rate ($Bq/m^2.s$)

Successive line 5's for input for each of the components indicated in line 4. Successive lines 3,4 and 5 are used for further branches with emanating components. The end is signalled by a blank line.

5.3 Program Output

Examples of the output for the radon level program are shown in Tables 5.5 and 5.8. The branch inputs are first listed in the output file. The branch emanation characteristics are then listed. Finally the calculated working levels, radon concentrations, equilibrium ratios and volumetric ages are listed for each branch.

5.4 Two Worked Examples

The two examples in this section are a continuation of the examples used for the ventilation program VENTPROG. The first example was the network used in both the basic concepts chapters 2.0 and 3.0. Input files for both examples were created using control code 5 of VENTPROG.

Example 1

The network diagram for the first example is shown in Figure 5.1. The branch input file is listed in Table 5.3 and the emanation characteristics in Table 5.4. The output for this example is shown in Table 5.5. Since the program is non-interactive, there is no output at the terminal.

Example 2

The network diagram for example 2 is shown in Figure 5.2 and the branch data and emanation characteristics input files are in Tables 5.6 and 5.7 respectively. The example provides an illustration of a simple recirculation loop (nodes 51,52 and 53). The output is detailed in Table 5.8.

FIGURE 5·1

NETWORK DIAGRAM FOR EXAMPLE 1

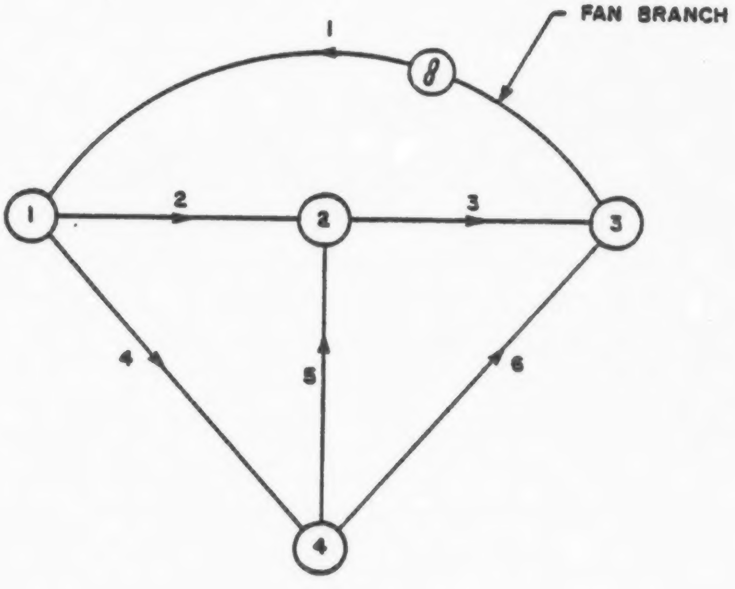

FAN BRANCH
1
8
2
3
1
2
3
4
5
6
4

Table 5.3

BRANCH FILE FOR EXAMPLE 1

```
          5         0.1    30        0
    3     1         3  0.57936  46.94126    1.00000    0.00000 748.00000    7.00000
    1     2         1  0.47485  21.98392    0.00000    0.00000 577.00000    7.00000
    2     3         1  0.84887  23.87164    0.00000    0.000001189.00000    7.00000
    1     4         1  0.36240  24.95734    0.00000    0.00000 393.00000    7.00000
    4     2         1  1.05004   1.88772    0.00000    0.000001528.00000    7.00000
    4     3         1  0.91609  23.06962    0.00000    0.000001299.00000    7.00000
0
```

Output from Ventilation Program (not needed)

Maximum Number of Iterations

Detailed printout control code

Branch Area (m^2

Branch Length (m)

Output from VENTPROG (not needed)

Balanced flows (m^3/s)

Output from VENTPROG (not needed)

Final Node

Initial Node

End of File

Table 5.4

EMANATION DATA FILE FOR EXAMPLE 1

```
         1
         0
  2.        1.
3
 500.     .46
 400.     .25
 400.     .72
  2.        3.
3
 500.     .46
 400.     .25
 400.     .72
  1.        4.
3
 500.     .46
 400.     .25
 400.     .72
  4.        2.
3
 500.     .46
 400.     .25
 400.     .72
  4.        3.
3
 800.     .46
 600.     .25
 500.     .72
  0.
```

number 1 initializies all radon and radon daughter levels in surface air intakes to zero

end of node initialization

initial and final node numbers for first branch

number of emanating components

3 emanating component areas (m^2) and respective emanation rates ($Bq/m^2.s$)

initial and final node numbers for second branch, etc.

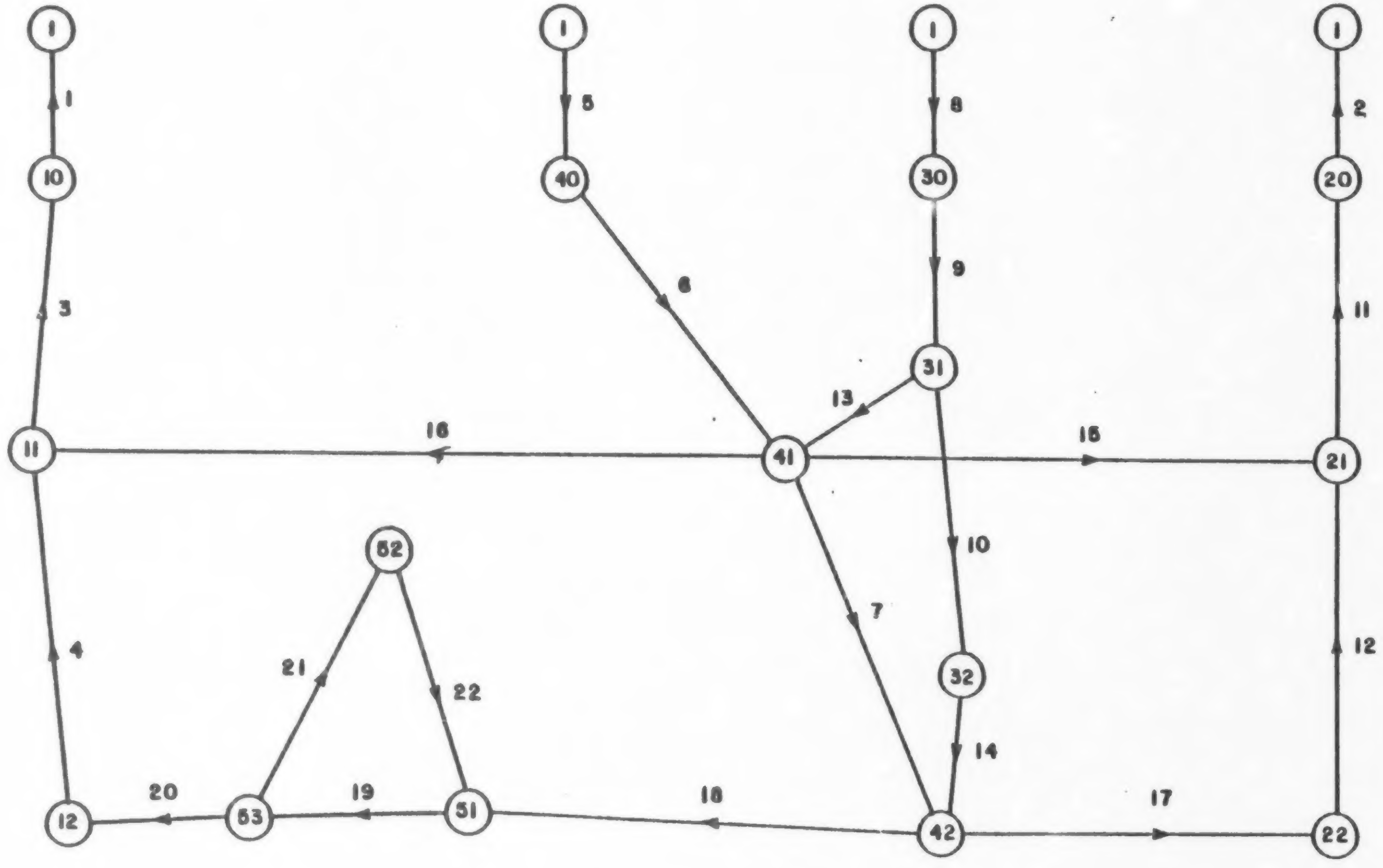
Table 5.5

Output for Example 1

```
     AIRWAYS          Q(M3/SEC)     X(M)    A(M2)

   1         2            22.0     577.0      7.0
   2         3            23.9    1189.0      7.0
   1         4            25.0     393.0      7.0
   4         2             1.9    1528.0      7.0
   4         3            23.1    1299.0      7.0


     AIRWAYS           AREA     EM RATE
                       (M2)     (BQ/(M2.SEC))


   1         2
                       500.0    0.460
                       400.0    0.250
                       400.0    0.720
   2         3
                       500.0    0.460
                       400.0    0.250
                       400.0    0.720
   1         4
                       500.0    0.460
                       400.0    0.250
                       400.0    0.720
   4         2
                       500.0    0.460
                       400.0    0.250
                       400.0    0.720
   4         3
                       800.0    0.460
                       600.0    0.250
                       600.0    0.720
1


INITIAL FINAL WORKING RADON   BRANCH     EQUILIBRIUM   RADIOACTIVE
 NODE   NODE    LEVEL CONC.   TRAVEL        RATIO        AGE(min)
                      (BQ/M3) TIME(min)
     1      2 0.0002    28.11     3.062        3.197       1.487
     2      3 0.0058    79.62     5.811       26.829      17.835
     1      4 0.0001    24.76     1.837        2.065       0.892
     4      2 0.0520   352.14    94.435       54.685      40.972
     4      3 0.0015    65.94     6.569        8.615       4.732
```

FIGURE 5·2

NETWORK DIAGRAM FOR EXAMPLE 2



1
1
1
1
1
5
8
2
10
40
30
20
6
9
11
3
31
13
16
15
11
41
21
10
52
7
4
12
21
22
32
14
20
19
18
17
12
53
51
42
22

Table 5.6
BRANCH INPUT FILE FOR EXAMPLE 2

```
      30   0.005           30        0
 10    1           3  0.00000  49.21056  1.00000  0.00000   0.00000   0.00000
 20    1           3  0.00000  49.83221  1.00000  0.00000   0.00000   0.00000
 10   11           1  0.01300 -49.21061  0.00000  0.00000 100.00000  10.00000
 11   12           1  0.01300 -19.75417  0.00000  0.00000 100.00000  10.00000
  1   40           1  0.00000  33.26971  0.00000  0.00000   0.00000   0.00000
 40   41           1  0.01700  33.26971  0.00000  0.00000 300.00000  15.00000
 41   42           1  0.01700  11.24603  0.00000  0.00000 300.00000  15.00000
  1   30           1  0.00000  65.77295  0.00000  0.00000   0.00000   0.00000
 30   31           1  0.00300  65.77295  0.00000  0.00000 100.00000  24.00000
 31   32           1  0.00300  31.60098  0.00000  0.00000 100.00000  24.00000
 20   21           1  0.01300 -49.83227  0.00000  0.00000 100.00000  10.00000
 21   22           1  0.01300 -23.09295  0.00000  0.00000 100.00000  10.00000
 31   41           1  0.00500  34.17196  0.00000  0.00000  50.00000  12.00000
 32   42           1  0.00500  31.60098  0.00000  0.00000  50.00000  12.00000
 41   21           1  0.05000  26.73931  0.00000  0.00000 500.00000  12.00000
 41   11           1  0.05000  29.45642  0.00000  0.00000 500.00000  12.00000
 42   22           1  0.05000  23.09290  0.00000  0.00000 500.00000  12.00000
 42   51           1  0.02000  19.75412  0.00000  0.00000 200.00000  12.00000
 51   53           1  0.01000  45.33205  0.00000  0.00000 100.00000  12.00000
 53   12           1  0.02000  19.75412  0.00000  0.00000 200.00000  12.00000
 53   52           1  0.01000  25.57798  0.00000  0.00000 100.00000  12.00000
 52   51           3  0.01000  25.57798  2.00000  0.00000 100.00000  12.00000
  0
```

Output From Ventilation Program (not needed)

Maximum Number of Iterations

Detailed Printout Code

Branch Area (m^2)

Branch Length (m)

Output from VENTRPOG (not neede

Balanced Flows (m^3/sec)

Output from VENTPROG (not neede

Final Node

Initial Node

End of Data

Table 5.7

EMANATION CHARACTERISTICS FILE FOR EXAMPLE 2

```
          1         number 1 initializes all radon and radon
                    daughter levels in surface air intakes to zero
          0
 10.     11.
3                   end of node initialization
 500.    .46
 400.    .25
 400.    .72        initial and final node numbers
 11.     12.        number of emanating components
3
 500.    .46        number of  emanating components and
 400.    .25
 400.    .72        3 emanating components area (m^2)
 20.     21.        emanation rate (Bq/m^2.s)
3
 500.    .46
 400.    .25
 400.    .72        initial and final node numbers
 21.     22.        etc.
3
 500.    .46
 400.    .25
 400.    .72
 30.     31.
3
 800.    .46
 600.    .25
 600.    .72
 31.     32.
3
 800.    .46
 600.    .25
 600.    .72
 40.     41.
1
4800.    .5
 41.     42.
1
4800.    .5
 31.     41.
3
 300.    .46
 200.    .25
 200.    .72
 32.     42.
3
 300.    .46
 200.    .25
 200.    .72
 42.     22.
3
```

Table 5.7 (continued)

```
3000.    .46
2000.    .25
2000.    .72
  41.    11.
3
3000.    .46
2000.    .25
2000.    .72
  51.    53.
3
2400.    .46
1600.    .25
1600.    .72
  53.    52.
3
2400.    .46
1600.    .25
1600.    .72
  52.    51.
3
 600.    .46
 400.    .25
 400.    .72
  52.    11.
3
 600.    .46
 400.    .25
 400.    .72
  53.    12.
3
1200.    .46
 800.    .25
 800.    .72
  42.    51.
3
1200.    .46
 800.    .25
 800.    .72
   0.
```

Table 5.8

OUTPUT FOR EXAMPLE 2

```
   AIRWAYS           Q(M3/SEC)     X(M)     A(M2)
11        10              49.2    100.0      10.0
12        11              19.8    100.0      10.0
 1        40              33.3      0.0       0.0
40        41              33.3    300.0      15.0
41        42              11.2    300.0      15.0
 1        30              65.8      0.0       0.0
30        31              65.8    100.0      24.0
31        32              31.6    100.0      24.0
21        20              49.8    100.0      10.0
22        21              23.1    100.0      10.0
31        41              34.2     50.0      12.0
32        42              31.6     50.0      12.0
41        21              26.7    500.0      12.0
41        11              29.5    500.0      12.0
42        22              23.1    500.0      12.0
42        51              19.8    200.0      12.0
51        53              45.3    100.0      12.0
53        12              19.8    200.0      12.0
53        52              25.6    100.0      12.0
52        51              25.6    100.0      12.0

   AIRWAYS            AREA     EM RATE
                      (M2)     (BQ/(M2.SEC))

11        10
                     500.0     0.460
                     400.0     0.250
                     400.0     0.720
12        11
                     500.0     0.460
                     400.0     0.250
                     400.0     0.720
21        20
                     500.0     0.460
                     400.0     0.250
                     400.0     0.720
22        21
                     500.0     0.460
                     400.0     0.250
                     400.0     0.720
30        31
                     800.0     0.460
                     600.0     0.250
                     600.0     0.720
31        32
                     800.0     0.460
                     600.0     0.250
                     600.0     0.720
40        41
                    4800.0     0.500
41        42
```

Table 5.8 (continued)

```
                             4800.0    0.500
       31       41
                              300.0    0.460
                              200.0    0.250
                              200.0    0.720
       32       42
                              300.0    0.460
                              200.0    0.250
                              200.0    0.720
       42       22
                             3000.0    0.460
                             2000.0    0.250
                             2000.0    0.720
       41       11
                             3000.0    0.460
                             2000.0    0.250
                             2000.0    0.720
       51       53
                             2400.0    0.460
                             1600.0    0.250
                             1600.0    0.720
       53       52
                             2400.0    0.460
                             1600.0    0.250
                             1600.0    0.720
       52       51
                              600.0    0.460
                              400.0    0.250
                              400.0    0.720
       53       12
                             1200.0    0.460
                              800.0    0.250
                              800.0    0.720
       42       51
                             1200.0    0.460
                              800.0    0.250
                              800.0    0.720

       ** SIMPLE RECIRCULATION LOOP:
    52  53  51
        NO. OF ITERATIONS FOR LOOP      11
1
```

Table 5.8 (continued)

INITIAL NODE	FINAL NODE	WORKING LEVEL	RADON CONC. (BQ/M3)	BRANCH TRAVEL TIME(min)	EQUILIBRIUM RATIO	RADIOACTIVE AGE(min)
11	10	0.0083	340.12	0.339	9.030	4.999
12	11	0.0164	576.62	0.844	10.502	5.964
1	40	0.0000	0.00	0.000	0.000	0.000
40	41	0.0005	72.14	2.254	2.460	1.095
41	42	0.0053	261.24	6.669	7.446	3.991
1	30	0.0000	0.00	0.000	0.000	0.000
30	31	0.0000	14.44	0.608	0.801	0.295
31	32	0.0003	44.51	1.266	2.096	0.908
21	20	0.0037	167.67	0.334	8.086	4.394
22	21	0.0061	279.67	0.722	8.108	4.409
31	41	0.0001	24.16	0.293	1.036	0.399
32	42	0.0003	55.01	0.316	2.278	1.000
41	21	0.0011	47.83	3.740	8.483	4.647
41	11	0.0021	160.54	3.395	4.818	2.400
42	22	0.0053	252.91	4.330	7.737	4.173
42	51	0.0030	176.37	2.025	6.291	3.278
51	53	0.0100	478.11	0.441	7.709	4.156
53	12	0.0144	545.34	2.025	9.770	5.481
53	52	0.0118	581.35	0.782	7.495	4.021
52	51	0.0138	607.31	0.782	8.395	4.591

REFERENCES

Archibald, J.F., and Nantel, J.H., 1984. "Prediction of Radiation Level and Ventilation Requirements in Underground Uranium Mines." CIM Bulletin, Vol. 77, No. 861, p. 60, January.

Austin, S.R., and Droullard, R.F., 1978. "Radon Emanation from Domestic Uranium Ores Determined by Modifications of the Close-Can, Gamma-Only Assay Method." U.S. Bureau of Mines Report of Investigations, No. 8264.

Chambers, D.B., Lowe, L.M., Cassaday, V.J., Nantel, J.H. and Archibald, J.F., 1980. "Development and use of Radon Source Terms in Environmental Impact Assessments." Health Physics Society, 25th Silver Anniversay Meeting, Seattle, Washington, July 20-25.

Cheng, K.C., 1981. "The Measurement of Radon Emanation Rates in a Canadian Cut-and-Fill Uranium Mine." CIM Bulletin, Vol. 74, No. 825, p. 110, January.

Evans, R.D., 1969. "Engineers' Guide to the Elementary Behaviour of Radon Daughters." Health Physics, Vol. 17, pp. 229-252.

Hartman, H.L., 1982. "Mine Ventilation and Air Conditioning." (Second Edition) (Co-editors: Mutmansky, J.M., Wang, Y.L.), John Wiley and Sons, Inc.

International Commission on Radiological Protection (ICRP), 1977. "Radiation Protection in Uranium and Other Mines." ICRP Publication 24, Annals of the ICRP, Vol. 1, No. 1.

Jacobi, W., 1972. "Activity and Potential Alpha Energy of Radon-222 and Radon-220 Daughters in Different Air Atmospheres." Health Physics, Vol. 22, pp. 441-450.

Jacobi, W., and Eisfeld, K., 1981. "Internal Dosimetry of Inhaled Radon Daughters." in Radiation Hazards in Mining: Control Measurement, and Medical Aspects, Gomez, M. (editor), Society of Mining Engineers, AIMME, New York.

Nielson, K.K., Perkins, R.W., Schwendiman, L.C. and Enderlin, W.I., 1979. "Prediction of the Net Radon Emission from a Model Open Pit Uranium Mine." Pacific Northwest Laboratory, PNL-2889 Rev., prepared for the U.S. NRC, NUREG/CR-0628.

Porstendorfer, J., Wicks, A. and Schraub, A., 1978. "The Influence of Exhalation, Ventilation and Deposition Processes Upon the Concentration of Radon (Rn-222), Thoron (Rn-220) and Their Decay Products in Room Air." Health Physics, Vol. 23, pp. 465-473.

Schroeder, G.L. and Evans, R.D., 1969. "Some Basic Concepts in Uranium Mine Ventilation." Transactions of the Society of Minng Engineers, AIME, Vol. 244, pp. 301-307.

Shearer, L.J., 1971. "An Introduction to System Dynamics." Addison-Wesley Publ. Co.

Tanner, A.B., 1964. "Radon Migration in the Ground: A Review." in The Natural Radiation Environment, Adams, J.S. and Lowder, W.M. (editors), University of Chicago Press.

Tanner, A.B., 1978. "Radon Migration in the Ground: A Supplementary Review." 3rd International Symposium on the Natural Radiation Environment, Houston, Texas, 23-28 April 1978.

Wheeler, A.J., 1982. "The Incorporation of Radon Growth Calculations Into Computer Ventilation Network Analysis." MSc. Thesis, Mining Engineering Dept., Queen's Univ.

APPENDIX A

TECHNICAL DESCRIPTION OF INDIVIDUAL MODULES FOR VENTPROG

APPENDIX A

TECHNICAL DESCRIPTION OF INDIVIDUAL MODULES FOR VENTPROG

VENTPROG - Main Program

Purpose

VENTPROG's Main Program essentially controls the other modules in the program. Other than several data validity checks no actual data processing takes place in this part of the program

Inputs

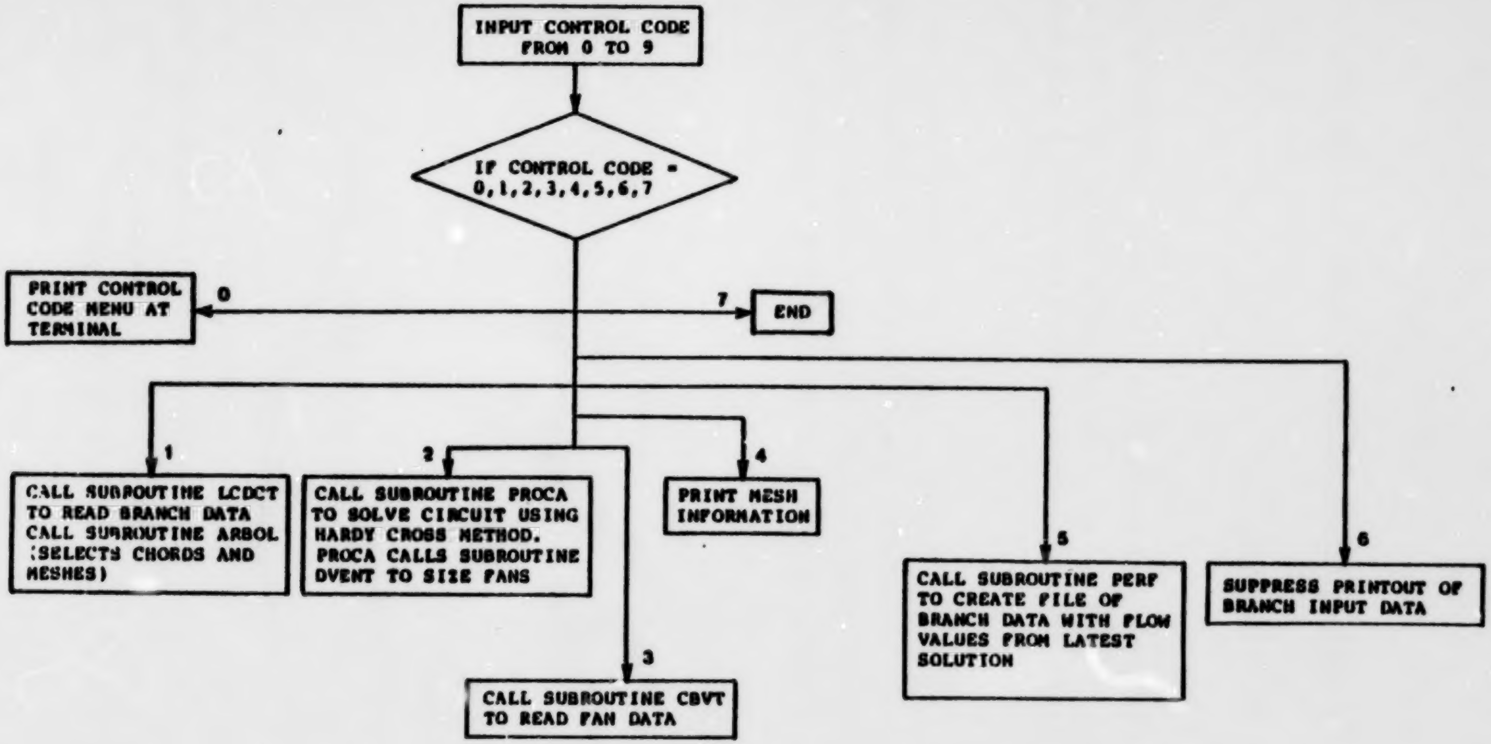
Inputs to the Main Program come in the form of control codes entered at the terminal (see Figure A.1). Other inputs come via the subroutines and are transferred to other subroutines via subroutine arguments or the COMMON BLOCK statement.

Variable Dictionary

Unless otherwise stated variables beginning with I,J,K,L,M and N are integers. All other variables are real. COMMOM BLOCK variables are also listed and denoted as CB. The terminal is denoted as "term".

Input/ Output	Source/ Destination	Integer Variables
		ICOMV - Data validity check variable
I	Term	IDIPR - Control code from 1 to 9
		IERS - Data validity check variable
		IMPR-I/O Control code
		IPER-I/O Control code

FIGURE A.1
FLOWCHART OF GENERAL STRUCTURE OF VENTPROG


INPUT CONTROL CODE FROM 0 TO 9
IF CONTROL CODE = 0,1,2,3,4,5,6,7
0
PRINT CONTROL CODE MENU AT TERMINAL
7
END
1
CALL SUBROUTINE LCDCT TO READ BRANCH DATA CALL SUBROUTINE ARBOL (SELECTS CHORDS AND MESHES)
2
CALL SUBROUTINE PROCA TO SOLVE CIRCUIT USING HARDY CROSS METHOD. PROCA CALLS SUBROUTINE DVENT TO SIZE FANS
3
CALL SUBROUTINE CBVT TO READ FAN DATA
4
PRINT MESH INFORMATION
5
CALL SUBROUTINE PERF TO CREATE FILE OF BRANCH DATA WITH FLOW VALUES FROM LATEST SOLUTION
6
SUPPRESS PRINTOUT OF BRANCH INPUT DATA

Input/ Output	Source/ Destination	
		IPLT-I/O Control code
		IRA - Control code indicating whether circuit has been changed
		JE-LOOP Limit
		JS-LOOP Limit
		K-LOOP Counter
I	term,CB	MAXIT-Maximum number of iterations
		MSGS-I/O Control Code
		LECT-I/O Control Code
0	LCTDT,CB	LIB-Number of chords, meshes and independent mesh equations
		MPRINT - Control code to suppress echo of input data
I	LCTDT,CB	NN- Number of nodes
I	LCTDT,CB	NFIX - Number of fixed branches
I	LCTDT,CB	NR - Number of branches
I	LCTDT,CB	NREG - Number of regulated branches
I	LCTDT,CB	NFAN - Number of fans
		NSIX - I/O Control code
		NSEVEN - I/O Control code
		NTV - Number of fans
		NTVP - Dummy variable - number of fans

Integer Arrays

I	LCTDT,CB	ITIPO (400) - Branch type
I	ARBOL,CB	ME(300) - Final branch in meshes
I	ARBOL,CB	NA(5000) - Array of branches in mesh order
I	LCTDT,CB	NNODO(300) - List of nodes
I	LCTDT,CB	NODOR(2,400) - Branch nodes
		NP(40) - No. of points on each fan curve

Input/ Output	Source/ Destination	
		Real Variables
		EP - Required precision for Hardy Cross correction factor
		XX - Dummy variable
		Real Arrays
I	LCTDT,CB	FLOW(400) - Branch flow
I	LCTDT,CB	CAUVE(400) - Additional information
I	CAERA,CB	FX(40,7) - Fan pressure
I	LCTDT,CB	PRES(400) - Natural ventilation pressures
I	LCTDT,CB	RES(400) - Branch resistances
		SUMPN(300) - Sum of pressures around mesh
		TIT(20) - Title
		VEN(40)- Fan numbers
		VENP(80)- Fan numbers associated with branches
		X(40,7) - Fan flow

Subroutine ARBOL

Purpose

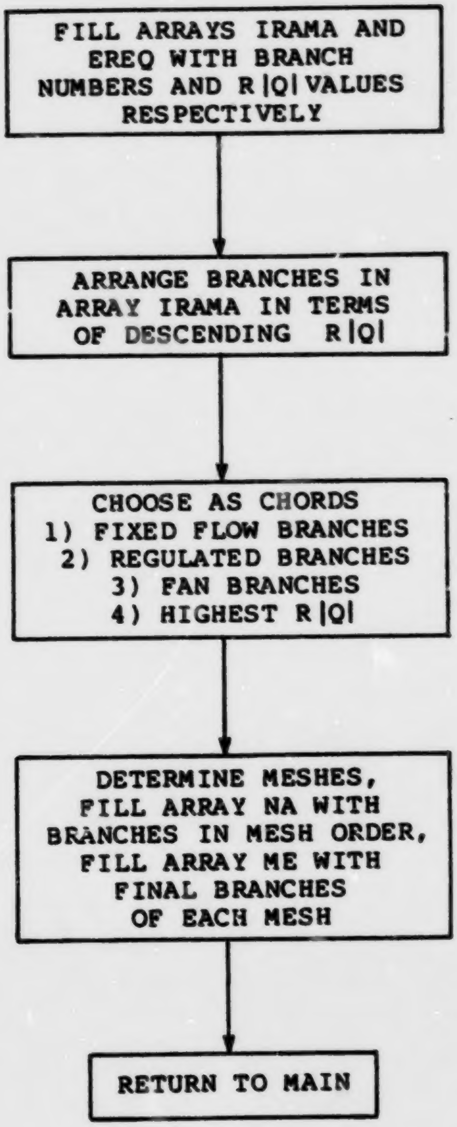
Subroutine ARBOL chooses the chords and sets up the meshes using the theories outlined in the Network Analysis section. Figure A.2 contains a general flowchart of subroutine ARBOL.

Inputs

Inputs to ARBOL come through the COMMON BLOCK statement.

FIGURE A.2

FLOWCHART FOR SUBROUTINE ARBOL


FILL ARRAYS IRAMA AND EREQ WITH BRANCH NUMBERS AND R|Q| VALUES RESPECTIVELY
ARRANGE BRANCHES IN ARRAY IRAMA IN TERMS OF DESCENDING R|Q|
CHOOSE AS CHORDS
1) FIXED FLOW BRANCHES
2) REGULATED BRANCHES
3) FAN BRANCHES
4) HIGHEST R|Q|
DETERMINE MESHES, FILL ARRAY NA WITH BRANCHES IN MESH ORDER, FILL ARRAY ME WITH FINAL BRANCHES OF EACH MESH
RETURN TO MAIN

Outputs

Outputs from ARBOL are transferred back to the main program through the COMMON BLOCK statement. ARBOL also sends the solved meshes with their branches listed to an output data set.

Variable Dictionary

Input/ Source/
Output Destination

Integer Variables

I - LOOP Counter
IE - Number of branches - 1 (loop limit)
II - Dummy variable
IJ - Loop counter
INE - Dummy variable
INS - Dummy variable
IS - Number of fixed and regulated branches + number of fans + 1 (loop limit)
J - Loop counter or dummy variable
JA - Dummy variable
JB - Dummy variable
JE - Dummy variable
JJ - Dummy variable
JX - Dummy variable
K - Dummy variable
KK - Dummy variable
K1 - Dummy variable (Loop limit)
L - Counter or dummy variable
L5 - Number of fixed and regulated branches - number of fans
M - Counter
MSGS- I/O Control code
N - Counter or dummy variable

Input/ Output	Source/ Destination	
		NN - Number of nodes
		NNSTP - Return code for abnormal termination
I	CB	NR - Number of Branches
I	CB	NRFIX - Number of fixed and regulated branches + 1 (loop limit)
I	CB	NFIX - Number of fixed branches
I	CB	NREG - Number of regulated branches
I	CB	NFAN - Number of fan branches
		Integer Arrays
		IRAMA(400) - Ordered array of branches
		JC-(300) - Status of nodes
		IVER(400)- Error codes (status of branches)
O	CB	NA(5000) - Array of branches in mesh order
I	CB	NODOR (2,400) - Branch nodes
I	CB	NNODO(300) - List of nodes
		Real variables
		T - Dummy variable
		Real Arrays
		FLOW(400) - Branch flow
		EREQ(400) - R Q

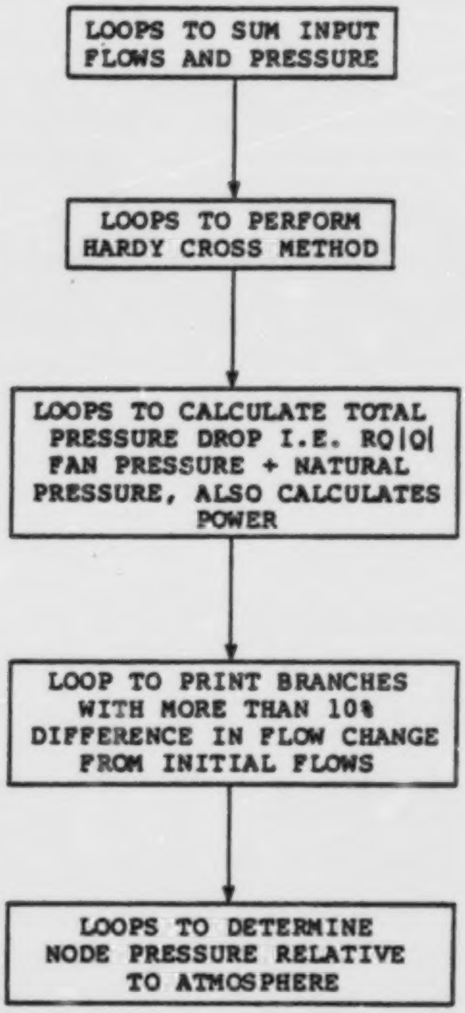
Subroutine PROCA

Purpose

The subroutine PROCA solves the mine network problem using the Hardy Cross method. The flowchart in Figure A.3 details the

FIGURE A.3

FLOWCHART FOR SUBROUTINE PROCA


LOOPS TO SUM INPUT FLOWS AND PRESSURE
LOOPS TO PERFORM HARDY CROSS METHOD
LOOPS TO CALCULATE TOTAL PRESSURE DROP I.E. RQ|Q| FAN PRESSURE + NATURAL PRESSURE, ALSO CALCULATES POWER
LOOP TO PRINT BRANCHES WITH MORE THAN 10% DIFFERENCE IN FLOW CHANGE FROM INITIAL FLOWS
LOOPS TO DETERMINE NODE PRESSURE RELATIVE TO ATMOSPHERE

general logic of the subroutine.

Inputs

The maximum specified number of iterations is passed from the main program to the subroutine via the subroutine arguments. The network information is sent via the COMMON BLOCK statements. Fan information is sent via the arguments for the subroutine DVENT.

Outputs

Detailed information on the solved networks is sent to an output data set. The COMMON BLOCK statement sends information on the solved network back to the main program

Variable Dictionary

Input/ Output	Source/ Destination	Integer Variables
		IBREF - Dummy variable
		IK - Dummy variable
		IIK - Dummy variable
		IXX - Dummy variable
		JE - Dummy variable
		JS - Dummy variable
		JS1 - Counter
		K - Dummy variable
		KK - Dummy variable
I	CB	LIB - number of chords, meshes and independent path equations
I	MAIN	MAXIT - Maximum number of iterations
		NRQ1 - Dummy variable

Input/ Output	Source/ Destination	
		N1 - Dummy variable
		N2 - Dummy variable
		Integer Arrays
I	CB	NA(5000) - Array of branches in mesh order
		NODES(300) - Node numbers
I	CB	NODOR(2,400) Branch nodes
I	CB	NNODO(300) - List of nodes
		Real Variables
		CADA1 - Dummy variable
		CAUV1 - Dummy variable
		CORR - Hardy Cross correction factor
		DDP - Denominator of Hardy Cross factor
O	DISK	DIFF - Difference between old and new flow
		DDPR - Slope of Q between two points on fan curve
		DEPRV - RES. Q
O	DISK	DEPRG - fan pressure drop
O	DISK	DEPRS - R |Q| Q
O	DISK	DEPTO - Total pressure drop in branch i.e. R |Q| Q + fan pressure drop
		DP - Numerator of Hardy Cross factor
		DPR - RES* |Q| Q
		POTT - Sum of power
O	DISK	POTG - Convergence index

Input/ Output	Source/ Destination	
O	DISK	POTVE - Total power requirement
O	DISK	POTRA - Power dissipated
O	DISK	SUMCO- Sum of correction factors for each mesh
O	DISK	TENTH- Criterion for highlighting changes (10% of previous flow)
		Real Arrays
I/O	CB/DISK	FLOW(400) - Branch flow
I	CB/DISK	CAUVE(400) - Additional information
O	DISK	DEPRE(400) - Pressure change in branch
		PRES(400) - Natural ventilation pressures
I/O	DISK	RES(400) - Branch resistance
		SUMPN(300) - Sum of pressures around mesh
O	DISK	STORE(400) - Previous flow value

Subroutine CBVT

Purpose

The subroutine reads and changes the fan characteristic curves.

Inputs

Fan data is read from disk file number 7. Information is transferred from the main program via the COMMON BLOCK statement.

Outputs

Fan data is sent to the output disk file. Fan data is sent to the main program via the COMMON BLOCK statements.

Variable Dictionary

Integer Variables

I-Loop counter
IERS - Error return code
IMPR-I/O Control Code
ITC - Fan elimination code
J - Loop counter
MSGS-I/O control code
NADA - Dummy read variable
NPP - Number of point on fan curve
NPV - Loop counter
Integer Arrays
NP(40)- Number of points on each fan curve

Real Arrays

FX(40,7) - Fan pressure
FXP(7) - Dummy array to ready fan pressure
VEN(40) - Fan numbers
X(40,7) - Fan flows
XP(7) - Dummy array to read fan flow characteristics

Subroutine DVENT

Purpose

The subroutine linearly interpolates fan pressure for a given flow. It also calculates the slope for that particular section of the curve.

Inputs

Data is sent from the subroutine PROCA via the COMMON BLOCK statement and subroutine arguments.

Outputs

Fan pressure drop and slope are sent back to subroutine PROCA via the subroutine arguments.

Variable Dictionary

Integer Variables

IVIC - Check variable to indicate if fan in extended range

INV - Additional information converted to integer

IK - Branch number

N - Number of numbers in fan characteristic data

Real Variables

DPR - Pressure drop

DDPR - Slope between 2 points on fan curve

Real Arrays

FLOW(400) - Branch flow

CAUVE(400) - Additional information

FX(40,7) Fan pressure data

X(40,7) - Fan flow data

Subroutine PERF

Purpose

The subroutine writes a complete set of branch data with the flow values generated by the latest solution into an output data set.

Inputs

Data is transferred via the COMMON BLOCK statement and the subroutine arguments

Output

The output is sent to a disk file.

Variable Dictionary

Integer Variables

IMPR-I/O Control code
K1 - Dummy variable

K2 - Dummy variable
MSGS - I/O Control Code
NR - Number of branches

Integer Arrays

ITIPO(400) - Branch type
NODOR(2,400) - Branch node

Real Arrays

FLOW(400) - Branch flow

CAUVE(400) - Additional information
RES(400) - Branch resistance

Subroutine LCTDT

Purpose

The subroutine reads a table of branch data from a disk file. It also calculates the number of fixed, fan and regulated branches and the number of branches and nodes.

Inputs

The inputs come from the disk file of branch data.

Output

The subroutine sends the branch data to the main program via the COMMON BLOCK statement

Variable Dictionary

Input/ Output	Source/ Destination	
		Integer Variables
		I-Loop counter
		MPRINT - If equal to 1, Print information, If equal to 2, don't print.
0	MAIN	NN - Number of nodes
0	MAIN	NREG - Number of regulated branches
0	MAIN	NFIX- Number of fixed branches
0	MAIN	NFAN - Number of fans

Integer Arrays

I	Disk	ITIPO(400) - Branch type
I	Disk	NODOR(2,400) Branch nodes

Real Arrays

I	Disk	FLOW (400) - Branch flow
I	Disk	CAUVE(400) - Additional information
I	Disk	PRES(400) - Natural ventilation pressure
I	Disk	RES(400) - Branch resistances

APPENDIX B

TECHNICAL DESCRIPTION OF INDIVIDUAL MODULES FOR VENTRAD

APPENDIX B

TECHNICAL DESCRIPTION OF INDIVIDUAL MODULES FOR VENTRAD

VENTRAD - MAIN program

Purpose

The MAIN program of VENTRAD handles the data input and output and follows the radon and radon daughters through the network. Figure B1 is a simplified flow chart of the MAIN program.

Inputs

Inputs to the MAIN program come via the branch input file (File 7) and the emanation characteristics file (File 8). The subroutine WORKL sends several parameters to MAIN via the COMMON block statement.

Outputs

The ouput from MAIN is sent to a disk file (File 6).

Variable Dictionary

Input/ Output	Source/ Destination	
		Integer Variables
		IFLAG - counter to find recirculation loops

Input/ Output	Source/ Destination	
I	disk (file 7)	IPR - number of iterations of detailed printout desired if recirculation loops are encountered (if IPR > ITER, all iterations will be printed)
I	disk (file 7)	ITER - maximum allowable number of iterations
		JE - end node of previous airway
		LFLAG - defines type of loop - if LFLAG=2 simple recirculation loop. If LFLAG>2 compound recirculation loop
		MPRINT - Printer control code
		NA - number of branches
I		NB - beginning node of branch (corresponds to emanation information)
		KLOOP - actual number of iterations
I	disk (file 8)	NE - end node of branch (corresponds to emanation information.
I	disk (file 8)	NK - number of different emanations
		Integer Arrays
I	disk (file 7)	IRAD (1,I) - Beginning node of branch 1
I	disk (file 8)	IRAD (2,I) - End node of branch I

Input/ Output	Source/ Destination	
		IRAD (3,1) - Logical variable (+/-) for branch I
		ISCAN (30) - Array that stores nodes of recirculation loops
		ILOOP (30) -
		ICOMB (20) - Array that stores branches that combine to form a node
I	disk (file 8)	NPAR (10) - Node numbers with known input parameters or filtration unit.
		Real variables
		XXX - Maximum allowed deviation between successive iterations
		Real Arrays
I	disk (file 8)	EM(1,K,I) - Area of the kth emanating element in the Ith branch
		EM(2,K,I) - Emanation rate corresponding to above
		PAR(1,I) - Known initial radon daughter concentration in the Ith branch. If PAR(1,I) < 0 then there is a filtration unit at node I
		PAR (2,I) - Radon daughter concentration at node I (WL)
I	disk (file 7)	RAD(1,I) - Flowrate (m^3/s) of branch I

Input/ Output	Source/ Destination	
I	disk (file 7)	RAD(2,I) - Airway length (m) of branch I
I	disk (file 7)	RAD(3,I) - Cross sectional area (m^2) of branch I
0	disk (file 6)	RAD(4,I) - Volumetric age (min) of air
		RAD(5,I) - Initial radon daughter concentration WLO
		RAD(6,I) - Initial radon concentration CO (Bq/m^3)
0	disk (file 6)	RAD(7,I) - Final radon daughter concentration WLO
0	disk (file 6)	RAD(8,I) - Final radon concentration CO (Bq/m^3)
		RAD(9,I) - Air residence time (min)
0	disk (file 6)	RAD(10,I) - Final equilibrium ratio (%)
		RAD(11,I) - Final radon concentration from previous iteration for comparison with the next iteration

Subroutine WORKL

Purpose

The subroutine is to compute the radon and radon daughter levels in an airway.

Inputs

Inputs to WORKL come via the subroutine arguments and the common block statement.

Outputs

Outputs from WORKL are transferred back to the MAIN program via the subroutine arguments and the common block statement.

Variable Dictionary

Input/ Output	Source/ Destination	
		Integer Variable
I	subroutine argument	L - Airway number
		MPRINT - Print control code
		Integer Arrays
I/O	CB/optional output disk (file 6)	IRAD(1,I) - Beginnig node of branch I
I/O	CB/optional output disk (file 6)	IRAD(2,I) - End node of branch I
I/O	CB/optional output disk (file 6)	IRAD(3,I) - Logical variable (+/-) for branch I
		Real Variables
		A - Equlibrium ratio at end of airway
		CA - Diffused radon added

Input/ Output	Source/ Destination	
		ER - Initial equilibrium ratio
		ES - Total radon emanation
		T - Radioactive age of air
		T - Branch residence time
		TT - Total area of air
		VOL - Branch volume
		WL1 - Working level growth from initial radon
		WL2 - WL growth from diffused radon
		Real Arrays
I	disk (file 8)	EM(1,K,I) - Area of the kth emanating element in the ith branch
		PAR(2,I) - Radon daughter concentration at node I (WL)
		RAD(1,I) - Flowrate (m^3/sec) of branch I
		RAD(2,I) - Airway length (m) of branch I
		RAD(3,I) - Cross-sectional area (m^2) of branch I
		RAD(4,I) - Radioactive age (min) of air
		RAD(5,I) - Initial radon daughter concentration WLO
		RAD(6,I) - Initial radon concentration CO (Bq/m^3)
		RAD(7,1) - Final radon daughter concentration WLO
I/O	CB/optional output disk (file 6)	RAD(8,I) - Final radon concentration CO (Bq/m^3)

Input/ Output	Source/ Destination	
		RAD(9,I) - Air residence time (min) RAD(10,I) - Final equilibrium ratio (%) RAD(11,I) - Final radon concentration (Bq/m^3) from previous iteration for comparison with the next iteration

APPENDIX C

PROGRAM LISTING - VENTPROG

```
C**************************MAIN PROGRAM********************************


C**************************MAIN PROGRAM********************************

C********************VARIABLE DICTIONARY******************************
C  UNLESS OTHERWISE STATED VARIABLES BEGINNING WITH I, J, K, L, M AND M
C  ARE INTEGERS. ALL OTHERS ARE REAL. COMMON BLOCK VARIABLES ALSO LISTED
C  INTEGER VARIABLES
C  ICOMV-DATA VALIDITY CHECK VARIABLE
C  ICCODE-CONTROL CODE FROM 1 TO 9
C  IERS-DATA VALIDITY CHECK VARIABLE
C  IMPR-I/O CONTROL CODE
C  IPER-I/O CONTROL CODE
C  IPLT-I/O CONTROL CODE
C  IRA-CONTROL CODE THAT INDICATES IF CIRCUT HAS BEEN CHANGED
C  JE-LOOP LIMIT
C  JS-LOOP LIMIT
C  K-LOOP COUNTER
C  MAXIT-MAXIMUM # OF ITERATIONS
C  MSGS-I/O CONTROL CODE
C  LECT-I/O CONTROL CODE
C  NMESH-NO. OF CHORDS, MESHES AND INDEPENDENT MESH EQUATIONS
C  MPRINT-
C  NN-NUMBER OF NODES
C  NFIX-NUMBER OF FIXED BRANCHES
C  NR-NUMBER OF BRANCHES
C  NRR-NUMBER OF REGULATED BRANCHES
C  NFAN-# OF FANS
C  NSIX-I/O CONTROL CODE
C  NSEVEN-I/O CONTROL CODE
C  NTV-NUMBER OF FANS
C  NTVP-DUMMY VARIABLE=# OF FANS
C  ____________________________________________________________________
C  INTEGER ARRAYS
C  ITIPO(400)-BRANCH TYPE
C  ME(300)-FINAL BRANCH IN MESHES LISTED IN NA(5000)
C  NA(5000)-ARRAY OF BRANCHES IN MESH ORDER
C  NNODO(300)-LIST OF NODES
C  NODOR(2,400)-BRANCH NODES
C  NP(40)-NUMBER OF POINTS ON EACH FAN CURVE
C  XX-DUMMY VARIABLE
C  ____________________________________________________________________
C  REAL VARIABLES
C  EP-REQUIRED PRECISION FOR HARDY CROSS CORRECTION FACTOR
C  ____________________________________________________________________
C  REAL ARRAYS
C  FLOW(400)-BRANCH FLOW
C  CAUVE(400)-ADDITIONAL INFORMATION
C  FX(40,7)-FAN PRESSURE
C  PRES(400)-NATURAL VENTILATION PRESSURES
C  RES(400)-BRANCH RESISTANCES
C  SUMPN(300)-SUM OF PRESSURES ARROUND MESH
C  TIT(20)-TITLE
C  VEN(40)-FAN NUMBERS
C  VENP(80)-FAN NUMBERS ASSOCIATED WITH BRANCHES
C  YCORD(400)-PLOTTING COORD.
C  XCORD(400)-PLOTTING COORD
C  X(40,7)-FAN FLOW
C  ________________________________________________________________
C  CHARACTER DATA
```

```
C*************************MAIN PROGRAM*********************************


C  HP-HORSEPOWER(HP)
C  WK-KILOWATTS(KW)
C  ****************************************************************
      DIMENSION VENP(80)
      COMMON RES(400),ITIPO(400),CAUVE(400),FLOW(400),PRES(400),
     *       NNODO(300),NODOR(2,400),NA(5000),ME(300),SUMPN(300),
     *     NTV,MPRINT,X(40,7),FX(40,7),NP(40),VEN(40),TIT(20),
     *       NR,NN,NFAN,NRR,NFIX,IERS,LECT,IMPR,IPER,NMESH,MSGS,IPLT,
     *     CHSZ,XCORD(400),YCORD(400),GRUNIT,GRIDWD,GRIDHT
     *,AREA(400),XLENGT(4)
C  LAST LINE OF COMMON STATEMENT CONTAINS PLOTTING OPTION VARS.
      DATA HP,WK/'HP','KW'/
C
C  INITIALIZE I/O FILE NUMBERS
      LECT=1
      IMPR=6
      IPER=9
      MSGS=1
C********************************************************************
C
C     FILE 5 IS A DISK FILE CONTAINING THE RUN PARAMETERS AND BRANCH
C      DATA.  RUN PARAMETERS ARE:
C        MAXIT:  MAXIMUM NUMBER OF ITERATIONS (NORMALLY 30)
C        EP:     PRECISION (.0001 ENGLISH OR .005 SI)
C     FILE 7 IS A DISK FILE CONTAINING FAN DATA
C     FILE 6 IS THE OUTPUT FILE (ROUTE TO DISK OR PRINTER)
C     FILE 1 CONSISTS OF CONTROL CODES ENTERED FROM THE TERMINAL
C     FILE 1 CONSISTS OF PROMPTS AND RESPONSE MESSAGES DIRECTED TO
C      THE TERMINAL
C     FILE 9 IS A DISK FILE GENERATED BY THE PROGRAM UNDER CONTROL
C      CODE 5 AND CONTAINING A BALANCED SOLUTION TO BE USED AS A
C      FILE 5 TO DETERMINE THE RELATIVE EFFECT OF SUBSEQUENT CHANGES
C      AS AN INPUT FILE TO THE RADON DAUGHTER PROGRAM
C
C*********************************************************************
      MPRINT=1
 2010 IERS=0
C
C  INPUT MAX. # ITERATIONS,PRECISION AND POWER COEFF
      READ(    5,2582)MAXIT,EP,ITERAD,IPRAD
       COEF=0.001
      IF(COEF.EQ.0.001)HP=WK
      WRITE(MSGS,2581)MAXIT,EP,HP,COEF
      NTV=0
2020  WRITE(MSGS,1000)
C
C  INPUT CONTROL CODES
2025  READ(LECT,*,ERR=2030)ICCODE
C  CHECK CONTROL CODE VALIDITY
      IF(ICCODE.GE.1.AND.ICCODE.LE.9)GO TO 2040
2030  WRITE(MSGS,1010)
      GO TO 2025
C  WRITE CONTROL CODE
2040  WRITE(IMPR,2570)ICCODE
      WRITE(MSGS,2570)ICCODE
C
C  GO TO APPROPRIATE PART OF PROGRAM AS DIRECTED BY CONTROL CODE
 2060 GO TO (2070,2100,2160,2170,2180,2185,2190),ICCODE
```

```
C***************************MAIN PROGRAM*******************************


C
 2070 CALL LCTDT  /*READS INITIAL CIRCUT DATA
      MPRINT=1
 2080 WRITE(IMPR,456)NN,NR,NFAN,NRR,NFIX /*WRITE NODE INFORMATION
      WRITE(MSGS,456)NN,NR,NFAN,NRR,NFIX
      NMESH=NR-NN+1 /*CALCULATE # OF MESHES
      CALL ARBOL /*SETS UP MESHES
      GO TO 2020

C  ******************************************************************
C  THIS SECTION CHECKS INPUT DATA AND THEN SEND TO SUBROUTINE PROCA
C  FOR HARDY CROSS SOLUTION
 2100 ICOMV=0
      IF(NTV)822,822,811
  811 NTVP=NTV
      DO 815 I=1,NR
      IF(ITIPO(I)-3)815,812,815
  812 DO 813 J=1,NTV
      IF(ABS(CAUVE(I))-VEN(J))813,814,813
  813 CONTINUE
      WRITE(MSGS,851)CAUVE(I)
      ICOMV=1
      NTVP=NTVP+1
      VENP(NTVP)=ABS(CAUVE(I))
      XX=FLOAT(NTVP)
      CAUVE(I)=SIGN(XX,CAUVE(I))
      GO TO 815
  814 XX=FLOAT(J)
      CAUVE(I)=SIGN(XX,CAUVE(I))
  815 CONTINUE
      IF(ICOMV+IERS)816,820,816
  816 DO 817 I=1,NTV
  817 VENP(I)=VEN(I)
      DO 819 I=1,NR
      IF(ITIPO(I)-3)819,818,819
  818 J=IFIX(ABS(CAUVE(I)))
      CAUVE(I)=SIGN(VENP(J),CAUVE(I))
  819 CONTINUE
  820 IF(ICOMV+IERS)2110,2140,2110
  822 IF(NFAN.EQ.0) GO TO 820
      WRITE(MSGS,852)
      ICOMV=1
 2110 WRITE(MSGS,2540)
      IF(ICOMV)2120,2130,2120
 2120 WRITE(MSGS,2550)
      IF(IERS)2130,2020,2130
 2130 WRITE(MSGS,2560)
      GO TO 2020
C  ***************************************************************
 2140 CALL PROCA (MAXIT,EP,COEF,HP) /*HARDY CROSS SOLUTION
      GO TO 2020
C  ****************************************************************
C  *****************************************************************
 2160 CALL CBVT /* READS AND CHANGES FAN CHARACTERISTIC CURVES
      MPRINT=1
      GO TO 2020
C  *****************************************************************
C WRITES MESH DATA
```

```
C************************MAIN PROGRAM*******************************

 2170 WRITE(IMPR,651)
      WRITE(MSGS,651)
      JE=0
      DO 601 I=1,NMESH
      JS=JE+1
      JE=ME(I)
  601 WRITE(IMPR,652)I,(NA(K),K=JS,JE)
      GO TO 2020
C  ***************************************************************
C  COMPLETE SET OF BRANCH DATA FROM 1ST SOLUTION IS WRITTEN TO FILE
 2180 WRITE(9,2582)MAXIT,EP,ITERAD,IPRAD
      WRITE(MSGS,2580)MAXIT,EP,COEF
      CALL PERF (ICOMV) /*CREATES NEW FILE
      GO TO 2020
C  ***************************************************************
C  SUPRESSES PRINTOUT OF BRANCH DATA
 2185 MPRINT=0
      GO TO 2020
C  ***************************************************************
C  END OF PROGRAM
 2190 STOP
C  ***************************************************************
C  FORMAT STATEMENTS
 456  FORMAT(/' CIRCUIT CHARACTERISTICS'/  ' ',23('-')//20X,'NODES', 10X
     * ,    I4/20X,'BRANCHES: TOTAL',I4/26X,'FAN ',I9/26X,'REGULATED',I4/
     *      26X,'FIXED',I8/)
  651 FORMAT(/' KIRCHOFF EQ. LOOPS '/' ',21('-')//' LOOP #     BRANCHES
     *'/)
  652 FORMAT(1H0,I4,8X,I4 /(16X,10I5))
  653 FORMAT('1')
  851 FORMAT(/' **ERROR** FAN CURVE #',F8.0,' MISSING'/)
  852 FORMAT(/' **ERROR** FAN CURVES MISSING'/)
 2520 FORMAT(20A4)
 2510 FORMAT(78X,I1,I1)
 2530 FORMAT(//20X,80(1H*)            //20X,20A4//20X,80(1H*)///)
 2540 FORMAT(/' ** RUN INTERRUPTED **')
 2550 FORMAT('   (DUE TO FAN ERROR)')
 2560 FORMAT('   (DUE TO CIRCUIT ERROR)')
 2570 FORMAT(/' ',20('*'),'CONTROL CODE #' ,I2, 60('*')/)
 2580 FORMAT(I10,2F10.5)
 2581 FORMAT(/' INITIAL DATA'/1X,11('-')//10X,'MAX.NUMBER OF ITER. ',I12/
     *10X,'PRECISION REQUIRED',F14.5/10X,'COEFFICIENT (P*Q TO )',A2,
     *F10.5)
2582  FORMAT(I10,F10.5,2I10)
 1000 FORMAT(' ENTER CONTROL CODE: '/' IF YOU WANT A MENU OF CODES,'
     *' TYPE "0" ')
 1010 FORMAT(' ENTER CONTROL CODE: '/' 1 = READ CIRCUIT DATA'/
     *' 2 = SOLVE CIRCUT'/
     *' 3 = READ FAN DATA'/' 4 = PRINT MESH DATA'/
     *' 5 = CREATE NEW BRANCH FILE'/' 6 = SUPPRESS PRINTOUT'/
     *' 7 = TERMINATE JOB')
      END
C  ***************************************************************
C  ***************************************************************

      SUBROUTINE ARBOL
C
C     THIS SUBROUTINE SETS UP THE MESHES OF THE NETWORK AS FOLLOWS:-
```

```
C**************************MAIN PROGRAM*******************************


C     CHOOSE AS CHORDS , THE BRANCHES OF TYPE 2, 4, AND
C      THE BRANCHES OF TYPE 3 WITH HIGHER VALUES OF R*!Q!.
C      THE MESHES ARE THE SET UP WITH ONE CHORD PER MESH.
C
C  **************************VARIABLE DICTIONARY**********************
C  SEE MAIN PROGRAM VARBLE DICTIONARY FOR COMMON BLOCK VARIABLES
C  INTEGER VARIABLES
C  I-LOOP COUNTER
C  IE-# OF BRANCHES-1(LOOP LIMIT)
C  II-DUMMY  VARIABLE
C  IJ-LOOP COUNTER
C  INE-DUMMY VARIABLE
C  INS-DUMMY VARIABLE
C  IS-FIXED+REG.+FAN+1(LOOP LIMIT)
C  J-LOOP COUNTER OR DUMMY VARIABLE
C  JA-DUMMY VARIABLE
C  JB-DUMMY VARIABLE
C  JE-DUMMY VARIABLE
C  JJ-DUMMY VARIABLE
C  JX-DUMMY VARIABLE
C  K-DUMMY VARIABLE
C  KK-DUMMY VARIABLE
C  K1-DUMMY VARIABLE(LOOP LIMIT)
C  L-COUNTER OR DUMMY VARIABLE
C  L5-# OF NORMAL+LINEAR CHARACTERISTIC BRANCHES
C  M-COUNTER
C  MSGS-I/O CONTROL CODE
C  N-COUNTER OR DUMMY VARIABLE
C  NN-NUMBER OF NODES
C  NNSTP-RETURN CODE FOR ABNORMAL TERMINATION
C  NR-NUMBER OF BRANCHES
C  NRFIX-FIXED+REQ.+1(LOOP LIMIT)
C  NFIX-NUMBER OF FIXED BRANCHES
C  NRR-NUMBER OF REGULATED BRANCHES
C  NFAN-NUMBER OF FAN BRANCHES
C  _______________________________________________________________
C  INTEGER ARRAYS
C  IRAMA(400)-ORDERED ARRAY OF BRANCHES
C  JC(300)-STATUS OF NODES
C  IUER(400)ERROR CODES(STATUS OF BRANCHES)
C  IT(200)
C  NA(5000)-ARRAY OF BRANCHES IN MESH ORDER
C  NODOR(2,400)-BRANCH NODES
C  NNODO(300)-LIST OF NODES
C _______________________________________________________________
C  REAL VARIABLES
C  T-DUMMY VARIABLE
C
C  _______________________________________________________________
C  REAL ARRAYS
C  FLOW(400)-BRANCH FLOW
C  EREG(400)-R!G!
C
C  *****************************************************************
      DIMENSION EREG(400),IRAMA(400),JC(300),IUER(400),IT(200)
      COMMON RES(400),ITIPO(400),CAUVE(400),FLOW(400),PRES(400),
     *      NNODO(300),NODOR(2,400),NA(5000),ME(300),SUMPN(300),
     *     NTV,MPRINT,X(40,7),FX(40,7),NP(40),VEN(40),TIT(20),
     *      NR,NN,NFAN,NRR,NFIX,IERS,LECT,IMPR,IPER,NMESH,MSGS,IPLT
```

```
C*************************MAIN PROGRAM******************************

C  ******************************************************************
C  CHECK TO SEE IF CIRCUT IS OPEN(VALIDITY CHECK)
C  --------------------BEGINNING OF LOOP----------------------------
      DO 2001 I=1,NN
 2001 JC(I)=0
C  -----------------------END OF LOOP-------------------------------
C  --------------------BEGINNING OF LOOPS---------------------------
      DO 2002 I=1,NR
      DO 2002 J=1,2
      K=NODOR(J,I)
 2002 JC(K)=JC(K)+1
C  -----------------------END OF LOOPS------------------------------
C  --------------------BEGINNING OF LOOPS---------------------------
      DO 2003 I=1,NN
      IF(JC(I)-1)2004,2004,2003
 2004 WRITE(MSGS,2101)NNODO(I)
 2003 CONTINUE
C  -----------------------END OF LOOPS------------------------------
C   *****************************************************************
C  THE FOLLOWING SECTION FILLS THE ARRAYS IRAMA AND EREG WITH
C  CONSECUTIVE BRANCH NUMBERS AND VALUES OF R¦Q¦ RESPECTIVELY
      J=0
      L = NFIX
      N = NFIX + NRR
      L5 = NFIX + NRR + NFAN
C
C -------------------------BEGINNING OF LOOP-----------------------
      DO 1034 I=1,NR
      K=ITIPO(I)
      GO TO (1035,1030,1031,1032,1035),K /*CHECK BRANCH TYPE
 1030 L=L+1
      M=L
      IUER(I)=0
      GO TO 1033
 1031 N=N+1
      M=N
      GO TO 1033
 1032 J=J+1
      IUER(I)=0
      M=J
      GO TO 1033
 1035 L5 = L5 + 1
      M = L5
 1033 IRAMA(M) = I
      EREG(M)=RES(I)*ABS(FLOW(I))
 1034 CONTINUE
C  -----------------------END OF LOOP-----------------------------

C  ******************************************************************
C  THE BRANCH ARRAY IRAMA(   ) IS FILLED WITH BRANCHES IN TERMS
C  OF DESCENDING R¦Q¦
      IS=NRR+NFAN+NFIX+1
      IE=NR-1
      JE=IE
C  --------------------BEGINNING OF LOOP----------------------------
      DO 1039 I=IS,IE
      L=0
C  --------------------BEGINNING OF LOOP----------------------------
```

```
      DO 1038 J=IS,JE
      IF(ABS(EREQ(J+1))-ABS(EREQ(J)))1038,1038,1037
 1037 T=EREQ(J)
      EREQ(J)=EREQ(J+1)
      EREQ(J+1)=T
      II=IRAMA(J)
      IRAMA(J)=IRAMA(J+1)
      IRAMA(J+1)=II
      L=1
 1038 CONTINUE
C  ----------------------------END OF LOOP--------------------------------
      IF(L)1040,1040,1039
 1039 JE=JE-1
C  ----------------------------END OF LOOP--------------------------------
C
C
C  ************************************************************************
C  DETERMINE CHORDS
C  ------------------------BEGINNING OF LOOP------------------------------
 1040 DO 1041 I=1,NN
 1041 JC(I)=0
C  ----------------------------END OF LOOP--------------------------------
      I=NR+1
      L=0
      N=0
      NRFIX = NFIX + NRR + 1
C  -------------------------BEGINNING OF LOOP-----------------------------
      DO 1052 IJ=NRFIX,NR
      I=I-1
      K=IRAMA(I)
      IUER(K) =1
      JA=NODOR(1,K)
      JB=NODOR(2,K)
      IF(JC(JA)-JC(JB))1047,1049,1042
 1042 IF(JC(JB))1043,1046,1043
 1043 JJ=JC(JB)
C  --------------------------BEGINNING OF LOOP---------------------------
      DO 1045 J=1,NN
      IF(JC(J)-JJ)1045,1044,1045
 1044 JC(J)=JC(JA)
 1045 CONTINUE
C  -----------------------------END OF LOOP-------------------------------
      GO TO 1052
 1046 JC(JB)=JC(JA)
      GO TO 1052
 1047 IF(JC(JA))1043,1048,1043
 1048 JC(JA)=JC(JB)
      GO TO 1052
 1049 IF(JC(JA))1051,1050,1051
 1050 L=L+1
      JC(JA)=L
      JC(JB)=L
      GO TO 1052
 1051 IUER(K)=0
      N=N+1
 1052 CONTINUE
C  ---------------------------------END OF LOOP---------------------------
      IF(N+NRR+NFIX-NMESH)1053,1054,1053
 1053 NNSTP=0
```

```
C***************************MAIN PROGRAM**********************************


      GO TO 1180
 1054 N=NMESH+1
C   -------------------------BEGINNING OF LOOP-------------------------
      DO 1058 I=NRFIX,NMESH
      K=IRAMA(I)
      IF(IUER(K))1055,1058,1055
C   -------------------------BEGINNING OF LOOP--------------------------
 1055 DO 1056 J=N,NR
      KK=IRAMA(J)
      IF(IUER(KK))1056,1057,1056
 1056 CONTINUE
C   ---------------------------END OF LOOP------------------------------
      NNSTP=1056
      GO TO 1180
 1057 II=IRAMA(I)
      IRAMA(I)=IRAMA(J)
      IRAMA(J)=II
      N=J+1
 1058 CONTINUE
C   ---------------------------END OF LOOP-------------------------------
C
C   ********************************************************************
C   USING CHORDS DETERMINE MESHES WITH 1 CHORD/MESH
      J =NMESH
      K=NMESH+1
      L=IRAMA(J)
      N=NODOR(1,L)
 1059 K1=K
      DO 1061 I=K1,NR
      L=IRAMA(I)
C ------------------------- BEGINNING OF LOOP------------------------
      DO 1061 M=1,2
      IF(N-NODOR(M,L))1061,1060,1061
 1060 IRAMA(I)=IRAMA(K)
      IRAMA(K)=L
      IF(M-1)1066,1066,1067
 1066 IUER(L)=1
      GO TO 1068
 1067 IUER(L)=-1
 1068 K=K+1
 1061 CONTINUE
C -------------------------- END OF LOOP-------------------------------
      IF(K-NR-1)1062,1070,1070
 1062 J=J+1
      L=IRAMA(J)
      IF(IUER(L))1063,1063,1064
 1063 M=1
      GO TO 1065
 1064 M=2
 1065 N=NODOR(M,L)
      GO TO 1059
 1070 M=0
C   -------------------------BEGINNING OF LOOP-------------------------
C   FILL ARRAY NA(3000) WITH BRANCHES IN MESH ORDER
      DO 1085 J=1,NMESH
      K=IRAMA(J)
      INE=NODOR(1,K)
      INS=NODOR(2,K)
```

```
C**************************MAIN PROGRAM*******************************


      M=M+1
      M1=0
      NA(M)=K
      JX=NR+1
 1071 JX=JX-1
      IF(JX-NMESH)1072,1073,1073
 1072 WRITE(IMPR,633)K
      WRITE(MSGS,633)K
      GO TO 1085
 1073 K=IRAMA(JX)
      IF(IUER(K))1076,1074,1075
 1074 NNSTP=K
      GO TO 1180
 1075 L=2
      N=1
      GO TO 1077
 1076 L=1
      N=2
 1077 IF(NODOR(L,K)-INS)1079,1078,1079
 1078 M=M+1
      NA(M)=-ISIGN(K,IUER(K))
      INS=NODOR(N,K)
      IF(INS-INE)1071,1081,1071
 1079 IF(NODOR(L,K)-INE)1071,1080,1071
 1080 M1=M1+1
      IT(M1)=ISIGN(K,IUER(K))
      INE=NODOR(N,K)
      IF(INE-INS)1071,1081,1071
 1083 M=M+1
      NA(M)=IT(M1)
      M1=M1-1
 1081 IF(M1)1084,1084,1083
 1084 ME(J)=M
 1085 CONTINUE
C  --------------------------END OF LOOP------------------------------
      RETURN
 1180 WRITE(IMPR,602)NNSTP
      WRITE(MSGS,602)NNSTP
      IERS=1
      RETURN
C  ******************END OF EXECUTABLE STATEMENTS*********************
  602 FORMAT(' **ERROR** CIRCUIT CANNOT BE SOLVED (',I3,')')
  633 FORMAT(/' BRANCH #',I5,' DOES NOT BELONG TO A LOOP'/)
 2101 FORMAT(/' **WARNING** CIRCUIT OPEN AT NODE',I6/)
      END
C  ***********************END OF SUBROUTINE****************************
C  *********************************************************************
C
      SUBROUTINE PROCA(MAXIT,EP,COEF,HP)
C
C     THIS SUBROUTINE BALANCES THE FLOWS TO GET AN INITIAL SOLUTION,
C      OBTAINS A BETTER APPROXIMATION BY THE HARDY-CROSS METHOD,
C      AND PRINTS THE SOLUTION
C
C  **************************VARIABLE DICTIONARY***********************
C  INTEGER VARIABLES
C  IBREG-DUMMY VARIABLE
C  IK-DUMMY VARIABLE
```

```
C***************************MAIN PROGRAM*********************************


C  IIK-DUMMY VARIABLE
C  INDIC-
C  IXX-DUMMY VARIABLE
C  JE-DUMMY VARIABLE
C  JS-DUMMY VARIABLE
C  JS1-COUNTER
C  K-DUMMY VARIABLE
C  KK-DUMMY VARIABLE
C  NMESH-# OF CHORDS , MESHES AND INDEPENDENT PATH EQUATIONS
C  MAXIT-MAXIMUM # OF ITERATIONS
C  NFIX1-DUMMMY VARIABLE
C  N1-DUMMY VARIABLE
C  N2-DUMMY VARIABLE
C_____________________________________________________________________
C   INTEGER ARRAYS
C   NA(5000)-ARRAY OF BRANCHES IN MESH ORDER
C   NODES(300)-NODE NUMBERS IN ASCENDING ORDER
C   NODOR(2,400)BRANCH NODES
C   NNODO(300)-LIST OF NODES
C   _____________________________________________________________________
C   REAL VARIABLES
C   CADA1-DUMMY VARIABLE
C   CAUV1-DUMMY VARIABLE
C   CORR- HARDY CROSS CORRECTION FACTOR
C   DDP-DENOMINATOR OF HARDY CROSS FACTOR
C   DDPR-SLOPE OF P VS Q BETWEEN 2 POINTS ON FAN CURVE
C   DDPRV-RES*!Q!
C   DP-NUMERATOR OF HARDY CROSS FACTOR
C   DPR-RES*!Q!*Q
C   POTT-SUM OF POWER
C   POTQ-CONVERGENCE INDEX
C   POTVE-TOTAL POWER REQUIREMENT
C   POTRA-POWER DISSIPATED
C   SUMCO-SUM OF CORRECTION FACTORS FOR EACH MESH
C   TENTH-CRITERION FOR HIGHLIGHTING CHANGES(10% OF PREVIOUS FLOW)
C   REAL ARRAYS
C   FLOW(400)-BRANCH FLOW
C   CAUVE(400)-ADDITIONAL INFORMATION
C   DEPRE(400)- PRESSURE CHANGE  IN BRANCH
C   PRES(400)-NATURAL VENTILATION PRESSURES
C   RES(400)-BRANCH RESISTANCE
C   SUMPN(300)-SUM OF PRESSURES ARROUND MESH
C   STORE(400)-PREVIOUS FLOW VALUE

      LOGICAL VN(400),V
      DIMENSION DEPRE(400),STORE(400),NODES(300)
      COMMON RES(400),ITIPO(400),CAUVE(400),FLOW(400),PRES(400),
     *       NNODO(300),NODOR(2,400),NA(5000),ME(300),SUMPN(300),
     *      NTV,MPRINT,X(40,7),FX(40,7),NP(40),VEN(40),TIT(20),
     *       NR,NN,NFAN,NRR,NFIX,IERS,LECT,IMPR,IPER,NMESH,MSGS
C   SEE MAIN PROGRAM VARIABLE DICTIONARY FOR COMMON BLOCK DEFINITIONS
C   ******************************************************************
C
C THE FOLLOWING LOOPS SUM INITIAL FLOWS AND PRESSURES
      JE=0
C  -------------------------BEGINNING OF LOOP-----------------------
      DO 1094 I=1,NMESH
      JS=JE+1
```

```
C**************************MAIN PROGRAM********************************


      JE=ME(I)
      K=IABS(NA(JS))
 1094 DEPRE(K)=FLOW(K)
C   ---------------------------------END OF LOOP----------------------------
C   -------------------------------BEGINNING OF LOOP------------------------
      DO 1095 I=1,NR
      STORE(I)=FLOW(I)
 1095 FLOW(I)=0.
C   ---------------------------------END OF LOOP----------------------------
      JE=0
C   -------------------------------BEGINNING OF LOOP------------------------
      DO 1000 I=1,NMESH
      JS=JE+1
      JE=ME(I)
      K=IABS(NA(JS))
      SUMPN(I)=PRES(K)
      FLOW(K)=DEPRE(K)
      JS1=JS+1
C   -------------------------------BEGINNING OF LOOP------------------------
      DO 1000 J=JS1,JE
      KK=IABS(NA(J))
      IF(NA(J))1099,1000,1098
 1098 FLOW(KK)=FLOW(KK)+DEPRE(K)
      SUMPN(I)=SUMPN(I)+PRES(KK)
      GO TO 1000
 1099 FLOW(KK)=FLOW(KK)-DEPRE(K)
      SUMPN(I)=SUMPN(I)-PRES(KK)
 1000 CONTINUE
C   ---------------------------------END OF LOOP----------------------------
C
C  ******************************************************************
C  LOOPS TO PERFORM HARDY CROSS METHOD
C  --------------------------------BEGINNING OF LOOP------------------------
      DO 1128 I=1,MAXIT
      NFIX1=NFIX+1
      JE=ME(NFIX)
      INDIC=0
      SUMCO=0.
C  --------------------------------BEGINNING OF LOOP------------------------
      DO 1127 L=NFIX1,NMESH
      IBREG=0
      JS=JE+1
      JE=ME(L)
      DDP=0.
      DP=SUMPN(L)
C  ----------------------------------BEGINNING OF LOOP----------------------
C  SUMMATION OF NUMERATOR AND DENOMINATOR IN HARDY CROSS CORR.FACTOR
      DO 1117 K=JS,JE
      IK=IABS(NA(K))
      IIK=ITIPO(IK)
      GO TO (1101,1102,1103,1100,1113),IIK
 1100 WRITE(IMPR,700)IK,K,J,I,NFIX,IIK
      NNSTP=1100
      WRITE(IMPR,700)NNSTP
      RETURN
 1101 DDPR=RES(IK)*ABS(FLOW(IK))
      DPR=DDPR*FLOW(IK)
      GO TO 1114
```

```
C*************************MAIN PROGRAM********************************


 1102 DDPR=RES(IK)*ABS(FLOW(IK))
      DPR=DDPR*FLOW(IK)
      IBREG=1
      CAUV1=CAUVE(IK)
      CADA1=FLOW(IK)
      GO TO 1114
 1103 CALL DVENT(IK,DDPR,DPR,NADA) /*CALL SUBPROGRAM TO SIZE FANS
 1112 DDPRV=RES(IK)*ABS(FLOW(IK))
      DPRV=DDPRV*FLOW(IK)
      DDPR=DDPRV-DDPR/2.
      DPR=DPRV-DPR
      GO TO 1114
 1113 DDPR=RES(IK)
      DPR=RES(IK)*FLOW(IK)
 1114 DDP=DDP+DDPR*2.
      IF(NA(K))1115,1116,1116
 1115 DP=DP+DPR
      GO TO 1117
 1116 DP=DP-DPR
 1117 CONTINUE
C  -------------------------END OF LOOP---------------------------------
C  CALCULATE CORRECTION FACTOR
      IF(ABS(DDP)-1.0E-20)1127,1127,1118
 1118 CORR=DP/DDP
      IF(IBREG)1121,1121,1119
 1119 IF((CAUV1-CADA1-CORR)*CAUV1)1120,1120,1121
 1120 CORR= CAUV1-CADA1
C  -------------------------BEGINNING OF LOOP---------------------------
C  ADD CORRECTION FACTOR TO PREVIOUS FLOW
 1121 DO 1124 K=JS,JE
      IK=IABS(NA(K))
      IF(NA(K))1123,1123,1122
 1122 FLOW(IK)=FLOW(IK)+CORR
      GO TO 1124
 1123 FLOW(IK)=FLOW(IK)-CORR
 1124 CONTINUE
C  -------------------------END OF LOOP---------------------------------
C  CHECK IF CORRECTION FACTOR IS REQUIRED PRECISION
      IF(ABS(CORR)-EP)1126,1126,1125
 1125 INDIC=1
 1126 SUMCO=SUMCO+ABS(CORR)
 1127 CONTINUE
C  -------------------------END OF LOOP---------------------------------
      IF (INDIC)1130,1130,1128
 1128 CONTINUE
C  -------------------------END OF LOOP---------------------------------
 1129 WRITE(IMPR,621) MAXIT,SUMCO
      WRITE(MSGS,621) MAXIT,SUMCO
      GO TO 1131
 1130 WRITE(IMPR,620)I,SUMCO
      WRITE(MSGS,620)I,SUMCO
 1131 JE=0
C  -------------------------BEGINNING OF LOOP---------------------------
      DO 1135 I=1,NMESH
      JS=JE+1
      JE=ME(I)
      DP=SUMPN(I)
      DDP=0.
```

```
C**************************MAIN PROGRAM*******************************

      IBREG=0
C   ------------------------BEGINNING OF LOOP----------------------
      DO 1149 L=JS,JE
      IK=IABS (NA(L))
      IIK=ITIPO(IK)
      GO TO (1132,1133,1134,1144,1145),IIK
 1132 DDPR=RES(IK)*ABS(FLOW(IK))
      DPR=DDPR*FLOW(IK)
      DEPRE(IK)=-DPR
      GO TO 1146
 1133 IBREG=1
      DDPR=RES(IK)*ABS(FLOW(IK))
      DPR=DDPR*FLOW(IK)
      CAUV1 =CAUVE(IK)
      CADA1=FLOW(IK)
      IK1=IK
      GO TO 1146
 1134 CALL DVENT(IK,DDPR,DPR,NADA)
 1143 DDPRV=RES(IK)*ABS(FLOW(IK))
      DPRV=DDPRV*FLOW(IK)
      DDPR=DDPRV-DDPR/2.
      DPR=DPRV-DPR
      DEPRE(IK)=-DPR
      GO TO 1146
 1144 IBREG=-1
      DPR=RES(IK)*FLOW(IK)*ABS(FLOW(IK))                              4
      DDPR=0.
      IK1=IK
      GO TO 1146
 1145 DDPR=RES(IK)
      DPR=RES(IK)*FLOW(IK)
 1146 DDP=DDP+DDPR*2.
      IF(NA(L))1147,1147,1148
 1147 DP=DP+DPR
      GO TO 1149
 1148 DP=DP-DPR
 1149 CONTINUE
C -----------------------------END OF LOOP---------------------------
      IF(IBREG)1152,1154,1150
 1150 IF(ABS(CADA1-CAUV1)-(EP/100.))1152,1152,1153
 1152 DEPRE(IK1)=-DP
      SUMPN(I)=0.
      GO TO 1155
 1153 DEPRE(IK1)=0.
 1154 SUMPN(I)=DP/DDP
 1155 CONTINUE
C   ---------------------- ---- END OF LOOP---------------------------
      WRITE(IMPP,624)HP
      POTC=0.
      POTT=0.
C
C   ****************************************************************
C   LOOPS TO CALCULATE TOTAL PRESSURE DROPS I.E.  G*R*|R|+NATURAL PRESS.
C   +FAN PRESS.   POWER ALSO CALCULATED
C   ---------------------BEGINNING OF LOOP---  ----------------------
      DO 1175 I=1,NR
      IIK=ITIPO(I)
      GO TO (1160,1161,1167,1166,160),IIK
```

```
C*****************************MAIN PROGRAM********************************

  160 DEPRS=-RES(I)*FLOW(I)
      GO TO 161
 1160 DEPRS=-RES(I)*FLOW(I)*ABS(FLOW(I))                                   5
  161 DEPRG=0.
      IK=1
      GO TO 1168
 1161 IK=2
 1162 DEPRG=DEPRE(I)
  162 DEPRS=-RES(I)*ABS(FLOW(I))*FLOW(I)                                   5
      POTVE=DEPRG*FLOW(I)*COEF
 1163 IF(POTVE)1165,1164,1164
 1164 POTT=POTT+POTVE
      GO TO 1168
 1165 IK=IK+1
      GO TO 1168
 1166 IK=4
      GO TO 1162
 1167 CALL DVENT(I,RNADA,DEPRG,IVIC) /*CALL SUBROUTINE TO SIZE FANS
      IK=6+IVIC
      IXX=IFIX(ABS(CAUVE(I)))
      CAUVE(I)=SIGN(VEN(IXX),CAUVE(I))
      GO TO 162
 1168 DEPTO=DEPRS+DEPRG+PRES(I)
      POTRA=-FLOW(I)*DEPRS*COEF
      POTG=POTG+DEPTO*FLOW(I)*COEF
      N1=NODOR(1,I)
      N2=NODOR(2,I)
      DEPRE(I)=DEPTO
      WRITE(IMPR,625)I,NNODO(N1),NNODO(N2),FLOW(I),DEPRS,DEPRG,POTRA,
     *DEPTO,IIK,RES(I),PRES(I),CAUVE(I)
      GO TO (1175,1169,1170,1171,1172,1173,1174,1174),IK
 1169 WRITE(IMPR,626)
      GO TO 1175
 1170 WRITE(IMPR,627)
      GO TO 1175
 1171 WRITE(IMPR,628) POTVE,HP
      GO TO 1175
 1172 WRITE(IMPR,629)
      GO TO 1175
 1173 WRITE(IMPR,630)POTVE,HP
      GO TO 1175
 1174 WRITE(IMPR,631)POTVE,HP
 1175 CONTINUE
C     -------------------------- END OF LOOP---------------------------
      IF(POTT.NE.0.) GO TO 1176
      WRITE(IMPR,634)
      WRITE(MSGS,634)
      RETURN
 1176 POTG=10000.*POTG/POTT
      WRITE(IMPR,632)POTT,HP,POTG
      IF(MPRINT.EQ.0)GO TO 1179
      WRITE(IMPR,635)
C     ****************************************************************
C     LOOP TO PRINT BRANCHES WITH MORE THAN 10% FLOW CHANGE
C     ---------------------BEGINNING OF LOOP--------------------------
      DO 1180 I=1,NR
      DIFF=FLOW(I)-STORE(I)
      TENTH=STORE(I)/10.
```

```
C**************************MAIN PROGRAM**********************************


      IF(ABS(DIFF). LT. ABS(TENTH))GO TO 1180
      N1=NODOR(1, I)
      N2=NODOR(2, I)
      WRITE(IMPR, 625)I, NNODO(N1), NNODO(N2), FLOW(I), STORE(I), DIFF
 1180 CONTINUE
 1179 MPRINT=1
C  ------------------------END OF LOOP----------------------------------
C  *********************************************************************
C  LOOPS TO DETERMINE NODE PRESSURE RELATIVE TO ATMOSPHERIC
C  ----------------------BEGINNING OF LOOP----------------------------
      DO 1181 I=1, NN
      VN(I)=. FALSE.
 1181 STORE(I)=0. 0
C  ------------------------END OF LOOP----------------------------------
      DO 1190 J=1, 10
      V=. TRUE.
      DO 1189 I=1, NR
      N1=NODOR(1, I)
      N2=NODOR(2, I)
      IF(NNODO(N1). NE. 1)GO TO 1182
      VN(N1)=. TRUE.
      STORE(N2)=DEPRE(I)
      VN(N2)=. TRUE.
      GO TO 1189
 1182 IF(NNODO(N2). NE. 1)GO TO 1183
      VN(N2)=. TRUE.
      STORE(N1)=-DEPRE(I)
      VN(N1)=. TRUE.
      GO TO 1189
 1183 IF(VN(N1))GO TO 1185
      IF(VN(N2))GO TO 1184
      V=. FALSE.
      GO TO 1189
 1184 STORE(N1)=STORE(N2)-DEPRE(I)
      VN(N1)=. TRUE.
      GO TO 1189
 1185 IF(VN(N2))GO TO 1189
      STORE(N2)=STORE(N1)+DEPRE(I)
      VN(N2)=. TRUE.
 1189 CONTINUE
C  ------------------------- END OF LOOP---------------------------------
      IF(V)GO TO 1191
 1190 CONTINUE
C  ------------------------- END OF LOOP---------------------------------
      WRITE(IMPR, 622)
      WRITE(MSGS, 622)
      RETURN
C  ------------------------BEGINNING OF LOOP---------------------------
 1191 DO 1195 I=1, NN
      NODES(I)=99999
      DO 1194 J=1, NN
      IF(. NOT. VN(J))GO TO 1194
      IF(NNODO(J). GT. NODES(I))GO TO 1194
      NODES(I)=NNODO(J)
      JE=J
 1194 CONTINUE
C  ------------------------END OF LOOP---------------------------------
      DEPRE(I)=STORE(JE)
```

```
C***********************MAIN PROGRAM***************************

      VN(JE)=.FALSE.
 1195 CONTINUE
C  ------------------------ END OF LOOP---------------------------
      WRITE(IMPR,623)(NODES(I),DEPRE(I),I=1,NN)
      RETURN
C  ***************END OF EXECUTABLE STATEMENTS*********************
  700 FORMAT(1H ,10I6)
  620 FORMAT(' NUMBER OF ITERATIONS',I5,/' LAST SUM OF ERRORS',F10.5)
  621 FORMAT(' PRECISION NOT REACHED AFTER ITERATION #',I6/20X,'LAST SUM
     * OF ERRORS',F10.5)
  622 FORMAT(//' ALL NODE PRESSURES NOT DETERMINED AFTER 10 ITERATIONS'
     */)
  623 FORMAT(1H1//' NODE PRESSURES RELATIVE TO ATMOSPHERIC',//
     * 1X,5(18H    NODE  PRESSURE)//5(I8,F10.3))
  624 FORMAT(//' RESULTS'/' ',7('-'),/'   BR. INI. FINAL  BRANCH     PRE
     *SS.',6X,'PRESS.',7X,'RES.',A2,6X,'PRESS.   BRANCH   RES.',7X,'NVP.
     *',4X,'ADD.'/'   NO. NODE NODE   FLOW',7X,'RESIS.       REG/FAN',19X,
     *'TOTAL',4X,'TYPE',23X,' DATA'/)
  625 FORMAT(' ',I5,I5,I5,5(F9.3,3X),I3,2X,F10.5,1X,F10.5,F7.2)
  626 FORMAT('+',108X,'UNRESTRICTED FLOW')
  627 FORMAT('+',108X,'RESTRICTED FLOW')
  628 FORMAT('+',108X,'FAN BRANCH',F7.2,1X,A2)
  629 FORMAT('+',108X,'REGULATOR BRANCH')
  630 FORMAT('+',108X,'FAN ',F7.2,1X,A2)
  631 FORMAT('+',108X,'**FAN OUT**',F7.2,1X,A2)
  632 FORMAT('0',     'TOTAL FAN POWER REQUIREMENT : ',F10.2,1X,A2/
     * 1X,'CONVERGENCE INDEX',F 7.1/)
  633 FORMAT(I10,2F10.4)
  634 FORMAT('0',10X,'TOTAL FAN POWER REQUIREMENT :  0.0'/)
  635 FORMAT(1H1//' BRANCHES WITH MORE THAN 10% FLOW CHANGE'//
     *'   BR. INI. FINAL   NEW',9X,'OLD',8X,'CHANGE',/
     *'   NO. NODE NODE    FLOW',8X,'FLOW'/)
      END
C  *************************************************************
C  *************************************************************
C  ************************************************************
C  ************************************************************
      SUBROUTINE CBVT
C
C     READS AND CHANGES FAN CHARACTERISTIC CURVES
C
C  ************************VARIABLE DICTIONARY**********************
C  INTEGER VARIABLES
C  I-LOOP COUNTER
C  IERS-ERROR RETURN CODE
C  IMPR-I/O CONTROL CODE
C  ITC-FAN ELIMINATION CODE
C  J-LOOP COUNTER
C  MSGS-I/O CONTROL CODE
C  NADA-DUMMY READ VARIABLE
C  NPP-NO. OF POINTS ON FAN CURVE
C  NPV-LOOP COUNTER
C  INTEGER ARRAYS
C  NP(40)-NO. OF POINTS ON EACH FAN CURVE
C  REAL VARIABLES
C  RE-DUMMY READ VARIABLE
C  REAL ARRAYS
C  FX(40,7)-FAN PRESSURES
```

```
C***************************MAIN PROGRAM*******************************


C  FXP(7)-DUMMY ARRAY TO READ FAN PRESSURES
C  VEN(40)-FAN NUMBERS
C  X(40,7)-FAN FLOWS
C  XP(7)-DUMMY ARAY TO READ FAN FLOW CHARACTERISTICS
C  ***********************************************************
      DIMENSION XP(7),FXP(7)
      COMMON RES(400),ITIPO(400),CAUVE(400),FLOW(400),PRES(400),
     *        NNODO(300),NODOR(2,400),NA(5000),ME(300),SUMPN(300),
     *       NTV,MPRINT,X(40,7),FX(40,7),NP(40),VEN(40),TIT(20),
     *        NR,NN,NFAN,NRR,NFIX,IERS,LECT,IMPR,IPER,NMESH,MSGS
      WRITE(IMPR,551)
      WRITE(MSGS,551)
  500 READ(    7,552)ITC,VENP,(XP(I),I=1,7)
      IF(ITC-1)508,501,508
C  ************************************************************
C  DATA VALIDITY CHECKS
C  ------------------BEGINNING OF LOOP--------------------------
  501 DO 502 I=1,NTV
      IF(VEN(I)-VENP)502,505,502
  502 CONTINUE
C  ---------------------END OF LOOP-----------------------------
      WRITE(IMPR,553)VENP
      WRITE(MSGS,553)VENP
      GO TO 500
  505 NPV=NP(NTV)
C  ---------------------BEGINNING OF LOOP-----------------------
      DO 506 J=1,NPV
      X(I,J)=X(NTV,J)
  506 FX(I,J)=FX(NTV,J)
C  ----------------------END OF LOOP----------------------------
      NP(I)=NP(NTV)
      WRITE(IMPR,554)VEN(I)
      WRITE(MSGS,554)VEN(I)
      VEN(I)=VEN(NTV)
      NTV=NTV-1
      GO TO 500
  508 IF(ITC+VENP+XP(1))510,509,510
  509 IF(NTV.EQ.0) RETURN
      WRITE(IMPR,558)(VEN(I),I=1,NTV)
      WRITE(MSGS,558)(VEN(I),I=1,NTV)
      RETURN
  510 IF(ITC)511,512,511
  511 WRITE(IMPR,556)VENP
      WRITE(MSGS,556)VENP
      GO TO 531
  512 IF(NTV-40)514,513,513
  513 WRITE(IMPR,555)VENP
      WRITE(MSGS,555)VENP
      GO TO 532
  514 IF (NTV)518,518,515
C  ----------------------------BEGINNING OF LOOP-----------------
  515 DO 517 I=1,NTV
      IF(VEN(I)-VENP)517,516,517
  516 WRITE(IMPR,559)VENP
      WRITE(MSGS,559)VENP
      GO TO 532
  517 CONTINUE
C  ------------------------------END OF LOOP--------------------
```

```
C***************************MAIN PROGRAM*********************************


  518 IF(XP(1)*XP(2))519,519,520
  519 READ(    7,557) NADA
      GO TO 530
C  -----------------------BEGINNING OF LOOP-----------------------
  520 DO 521 I=3,7
      IF(XP(I))521,522,521
  521 CONTINUE
C  -------------------------END OF LOOP----------------------------
      NPP=7
      GO TO 523
  522 NPP=I-1
  523 READ(    7,552)NADA,RE,(FXP(I),I=1,NPP)
C  -----------------------BEGINNING OF LOOP-----------------------
C  DATA CHECK-CHECK FLOWS FOR ASCENDING ORDER,PRESSURES FOR
C  DESCENDING
      DO 527 I=2,NPP
      IF(XP(I)-XP(I-1))530,530,526
  526 IF(FXP(I)-FXP(I-1))527,527,530
  527 CONTINUE
C  -------------------------END OF LOOP----------------------------
      NTV=NTV+1
C  -----------------------BEGINNING OF LOOP-----------------------
C  FAN FLOW AND PRESSURE INFORMATION SUBSTITUTED INTO X(40,7)&
C  FX(40,7)
      DO 528 I=1,NPP
      X(NTV,I)=XP(I)
  528 FX(NTV,I)=FXP(I)
C  -------------------------END OF LOOP----------------------------
      NP(NTV)=NPP
      VEN(NTV)=VENP
      IF(MPRINT.EQ.0)GO TO 500
      WRITE(IMPR,560)VEN(NTV),(X(NTV,I),FX(NTV,I),I=1,NPP)
      WRITE(IMPR,552)
      GO TO 500
  530 IERS=1
      WRITE(IMPR,556)VENP
      WRITE(MSGS,556)VENP
      GO TO 500
  531 IERS=1
  532 READ(    7,557) NADA
      GO TO 500
C  *****************END OF EXECUTABLE STATEMENTS*******************
  551 FORMAT(/' FAN CHARACTERISTIC CURVES'/1X,25('-')//)
  552 FORMAT(I1,F9.0,7F10.4)
  553 FORMAT(/' **ERROR** FAN #',F8.0,' DOES NOT EXIST'/)
  554 FORMAT (  /,' FAN CURVE #',F8.0,' ELIMINATED' )
  555 FORMAT(/' **ERROR** MAX.ACCEPTED: 40 CURVES')
  556 FORMAT(/' **ERROR** IN CURVE # ',F8.0 )
  557 FORMAT(1I1)
  558 FORMAT(/' FAN  CURVES IN : '//(5X,F8.0))
  559 FORMAT(/' CURVE # ',F8.0,' READ TWICE')
  560 FORMAT(13X,'CURVE #',F5.0,6X,'FLOW',8X,'PRESSURE'//(26X,F9.3,5X,F
     *9.3/))
      END
C  ****************************************************************
C  ****************************************************************
      SUBROUTINE DVENT(IK,DDPR,DPR,IVIC)
C
```

```
C**************************MAIN PROGRAM*******************************


C     CALCULATES FAN PRESSURE FOR A GIVEN FLOW, AND SLOPE OF CURVE
C
C  *************************VARIABLE DICTIONARY***********************
C  INTEGER VARIABLES
C  IVIC-CHECK VARIABLE TO INDICATE IF FAN IN EXTENDED RANGE
C  INV- ADDITIONAL INFORMATION CONVERTED TO INTEGER
C  IK-BRANCH NO.
C  N-NUMBER OF NUMBERS IN FAN CHARACTERISTIC DATA
C  NP-
C  REAL VARIABLES
C  DPR-PRESSURE DROP
C  DDPR-SLOPE BETWEEN 2 POINTS ON FAN CURVE
C  REAL ARRAYS
C  FLOW(400)-BRANCH FLOW
C  CAUVE(400)-ADDITIONAL INFORMATION
C  FX(40,7)FAN PRESSURE DATA
C  X(40,7)-FAN FLOW DATA
C  **************************************************************
      COMMON RES(400),ITIPO(400),CAUVE(400),FLOW(400),PRES(400),
     *       NNODO(300),NODOR(2,400),NA(5000),ME(300),SUMPN(300),
     *      NTV,MPRINT,X(40,7),FX(40,7),NP(40),VEN(40),TIT(20),
     *       NR,NN,NFAN,NRR,NFIX,IERS,LECT,IMPR,IPER,NMESH,MSGS
      IVIC=0
 1134 INV=IFIX(ABS(CAUVE(IK)))
      ABCAU=FLOW(IK)*SIGN(1.,CAUVE(IK))
      IF(X(INV,1)-ABCAU)1136,1135,1135 /*CHECK BRANCH FLOW AGAINST
C  FAN FLOW
 1135 DPR=FX(INV,1) /*USE LOWER EXTENDED RANGE
      DDPR=0.
      IVIC=1
      GO TO 1141
 1136 N=NP(INV)
C  ------------------------BEGINNING OF LOOP-----------------------
C  USE NORMAL OR UPPER EXTENDED RANGE
      DO 1137 IJ=2,N
      IF(X(INV,IJ)-ABCAU)1137,1138,1138
 1137 CONTINUE
C  -------------------------END OF LOOP----------------------------
      DDPR=0.
      DPR=FX(INV,N)
      IVIC=1
      GO TO 1141
C  ****************************************************************
C  LINEAR INTERPOLATION BETWEEN 2 POINTS IN NORMAL OPERATING
C  RANGE
 1138 DDPR=(FX(INV,IJ)-FX(INV,IJ-1))/(X(INV,IJ)-X(INV,IJ-1))
      DPR=FX(INV,IJ)-DDPR*(X(INV,IJ)-ABCAU)
C  ****************************************************************
 1141 IF(CAUVE(IK))1142,1143,1143
 1142 DPR=-DPR
 1143 RETURN
      END
C  ******************************************************************
C  ******************************************************************
      SUBROUTINE PERF(ICOMV)
C
C     CREATES A NEW FILE OF BRANCH DATA WITH THE BALANCED FLOW
C      DISTRIBUTION FROM THE LAST SOLUTION
```

```
C**************************MAIN PROGRAM*******************************


C
C    **************************VARIABLE DICTIONARY*********************
C    INTEGER VARIABLES
C    ICOMV-
C    IERS-
C    IMPR-I/O CONTROL CODE
C    K1-DUMMY VARIABLE
C    K2-DUMMY VARIABLE
C    MSGS-I/O CONTROL CODE
C    NR-NUMBER OF BRANCHES
C    INTEGER ARRAYS
C    ITIPO(400)-BRANCH TYPE
C    NODOR(2,400)-BRANCH NODE
C    REAL ARRAYS
C    FLOW(400)-BRANCH FLOW
C    CAUVE(400)-ADDITIONAL INFORMATION
C    RES(400)-BRANCH RESISTANCES
C    ************************************************************
      COMMON RES(400),ITIPO(400),CAUVE(400),FLOW(400),PRES(400),
     *        NNODO(300),NODOR(2,400),NA(5000),ME(300),SUMPN(300),
     *       NTV,MPRINT,X(40,7),FX(40,7),NP(40),VEN(40),TIT(20),
     *        NR,NN,NFAN,NRR,NFIX,IERS,LECT,IMPR,IPER,NMESH,MSGS
     *,XLENG1(400),AREA(400)
      IF(ICOMV+IERS)701,702,701
  701 WRITE(IMPR,750)
      WRITE(MSGS,750)
  702 DO 705 I=1,NR
      K1=NODOR(1,I)
      K2=NODOR(2,I)
  705 WRITE(IPER,756)NNODO(K1),NNODO(K2),ITIPO(I),RES(I),FLOW(I),CAUVE
     *(I),PRES(I),XLENG1(I),AREA(I)
      WRITE(IPER,753)
      RETURN
C    *********************END OF EXECUTABLE STATEMENTS*****************
  750 FORMAT(//' DATA SET NOT PUNCHED DUE TO ERROR IN DATA')
  751 FORMAT(78X,'11'/20A4)
  752 FORMAT(80X)
  753 FORMAT(79X,'2')
  754 FORMAT(1F10.0,7F10.4)
  755 FORMAT(10X,7F10.4)
  756 FORMAT(2I5,I10,6F10.5)
      END
C    ************************************************************
C    ************************************************************
      SUBROUTINE LCTDT
C
C     READS THE INITIAL DATA OF THE CIRCUIT
C
C    ***************************VARIABLE DICTIONARY******************
C    SEEMAIN PROG. VARIABLE DICTIONARY FOR LISTING OF COMMON BLOCK
C    VARIABLES
C    INTEGER VARIABLES
C    I-LOOP COUNTER
C    MPRINT-IF 1 PRINT INFORMATION, IF 0 DON'T PRINT
C    NN-NUMBER OF NODES
C    NRR-NUMBER OF REGULATED BRANCHES
C    NFIX-NUMBER OF FIXED BRANCHES
C    NFAN-NUMBER OF FANS
```

```
C**************************MAIN PROGRAM*******************************


C   INTEGER ARRAYS
C   ITIPO(400)-BRANCH TYPE
C   NODOR(2,400)BRANCH NODES
C   REAL ARRAYS
C   FLOW(400)-BRANCH FLOW
C   CAUVE(400)-ADDITIONAL INFORMATION
C   PRES(400)-NATURAL VENTILATION PRESSURE
C   RES(400)-BRANCH RESISTANCES
C   **************************************************************
        COMMON RES(400),ITIPO(400),CAUVE(400),FLOW(400),PRES(400),
     *        NNODO(300),NODOR(2,400),NA(5000),ME(300),SUMPN(300),
     *       NTV,MPRINT,X(40,7),FX(40,7),NP(40),VEN(40),TIT(20),
     *        NR,NN,NFAN,NRR,NFIX,IERS,LECT,IMPR,IPER,NMESH,MSGS
     *,XLENG1(400),AREA(400)
        DIMENSION PERIM(400),RESFAC(400),XLENG2(400)
      NN=1
      NRR=0
      NFIX=0
      NFAN=0
      IF(MPRINT.EQ.1)WRITE(IMPR,251)
      DO 210 I=1,401
      READ(5,252)NODOR(1,I),NODOR(2,I),ITIPO(I),RES(I),FLOW(I),CAUVE(I),
     *PRES(I),XLENG1(I),AREA(I)
      IF(NODOR(1,I)+NODOR(2,I)+ITIPO(I))211,211,201
 201  IF(RES(I).EQ.-1.0)READ(5,255)XLENG1(I),XLENG2(I),AREA(I),PERIM(I)
     *,RESFAC(I)
      IF(RES(I).EQ.-1.0)RES(I)=(RESFAC(I)*PERIM(I)*(XLENG1(I)+XLENG
     *2(I)))/AREA(I)**3
C   SUBROUTINE IF LAST RECORD READ
      IF(MPRINT.EQ.0)GO TO 260
      WRITE(IMPR,253)I,NODOR(1,I),NODOR(2,I),ITIPO(I),RES(I),FLOW(I),
     *CAUVE(I),PRES(I)
C *******************************************************************
C   COUNT BRANCH TYPES
  260 K=ITIPO(I)
      GO TO (204,202,203,205,204),K
  202 NRR=NRR+1
      GO TO 204
  203 NFAN=NFAN+1
      GO TO 204
  205 NFIX=NFIX+1
C  ******************************************************************
  204 IF(I-1)206,206,207
  206 NNODO(1)=NODOR(1,1)
C ------------------------BEGINNING OF LOOP-------------------------
  207 DO 210 L=1,2
C  -----------------------BEGINNING OF LOOP-------------------------
      DO 209 J=1,NN
      IF(NNODO(J)-NODOR(L,I))209,208,209
  208 NODOR(L,I)=J
      GO TO 210
  209 CONTINUE
C  --------------------------END OF LOOP----------------------------
      NN=NN+1 /*COUNT # OF NODES
      NNODO(NN)=NODOR(L,I)
      NODOR(L,I)=NN
  210 CONTINUE
C  --------------------------END OF LOOP-----------------------------
```

```
C***************************MAIN PROGRAM*********************************


      IERS=1
      WRITE(IMPR ,254)
      WRITE(MSGS ,254)
  211 NR=I-1 /*COUNT # OF BRANCHES
      RETURN
C   ********************END OF EXECUTABLE STATEMENTS*******************
  253 FORMAT(2X,I5,3X,3(I5,1X),F13.5,3X,3(F12.3,4X))
  251 FORMAT(   1X,'DATA READING'/ 1X,12('-')//1X,'BRANCH #',2X,'IN. ',2X,
     *'FIN. ',    4X,'TYPE',4X,'RESISTANCE',10X,'FLOW',10X,'ADD.DATA',7X,
     *'NAT.PRESS. '/)
  252 FORMAT(2I5,I10,6F10.4)
  254 FORMAT(' **ERROR** MAX.CAPACITY :  400 BRANCHES')
 255  FORMAT(5F10.4)
      END
C   ******************************************************************
C   ******************************************************************
      SUBROUTINE PDRIVR
C
C   ENDS PROGRAM
      RETURN
      END
C   ******************************************************************
C   ******************************************************************
```

APPENDIX D

PROGRAM LISTING - VENTRAD

```

C*******************************************************************
C                        RADON NETWORK PROGRAM
C*******************************************************************
C
C              WRITTEN  BY ADAM JULIAN WHEELER  JUNE 1982
C
C              MODIFIED BY KEITH R. NOTLEY  AUGUST 1984
C
C*******************************************************************
C
C    IRAD(1,I) = BEGINNING NODE OF BRANCH I
C    IRAD(2,I) = END       NODE OF BRANCH I
C    IRAD(3,I) = LOGICAL VARIABLE  OF BRANCH I (+/-)
C    RAD(1,I)  = FLOWRATE (M3/SEC) OF BRANCH I
C    RAD(2,I)  = AIRWAY LENGTH        OF BRANCH I (M)
C    RAD(3,I)  = CROSS-SECTIONAL AREA OF BRANCH I (M2)
C    RAD(4,I)  = VOLUMETRIC AGE OF AIR
C    RAD(5,I)  = INITIAL RADON DAUGHTER CONCENTRATION WLO
C    RAD(6,I)  = INITIAL RADON CONCENTRATION  CO (BQ/M3)
C    RAD(7,I)  = FINAL   RADON DAUGHTER CONCENTRATION WLO
C    RAD(8,I)  = FINAL   RADON CONCENTRATION  CO (BQ/M3)
C    RAD(9,I)  = AIR RESIDENCE TIME (MINUTES)
C    RAD(10,I) = FINAL EQUILIBRIUM RATIO (%)
C    RAD(11,I) = FINAL RADON CONCENTRATION FROM PREVIOUS ITERATION
C                FOR COMPARISON WITH THE NEXT ITERATION
C
C    ISCAN(I)  \  ARRAYS THAT STORE NODES OF RECIRCULATION LOOPS
C    ILOOP(I) /
C
C    ICOMB(I) = ARRAY THAT STORES THE BRANCHES COMBINED INTO A NODE
C
C
C    XXX       = MAXIMUM ALLOWED DEVIATION BETWEEN SUCCESSIVE ITERATI
C                (BQ/M3)
C
C
C
C*******************************************************************
C
      DIMENSION NPAR(10),PAR(2,10),ISCAN(30),ICOMB(20),ILOOP(30)
      COMMON IRAD(3,70),MPRINT,RAD(11,70),EM(2,5,70)
C-------------------------------------------------------------------
C     DEFINE MAXIMUM ALLOWED DEVIATION BETWEEN SUCCESSIVE ITERATIONS
C-------------------------------------------------------------------
      XXX=1.
C-------------------------------------------------------------------
C                INITIALIZE RAD AND EM ARRAY VARIABLES
C-------------------------------------------------------------------
      DO 10 J=1,70
        DO 6 I=1,3
          IRAD(I,J)=0
    6   CONTINUE
        DO 5 I=1,10
          RAD(I,J)=0.0
    5   CONTINUE
        RAD(11,J)=-100.
        DO 8 K=1,5
```

```
              EM(1,K,J)=0.0
              EM(2,K,J)=0.0
    8     CONTINUE
   10  CONTINUE
       DO 9 J=1,30
          ILOOP(J)=0
          ISCAN(J)=0
    9  CONTINUE
       ILL=0
       LFLAG=0
       KLOOP=0
       LOOP=0
       WRITE(6,2)
    2  FORMAT(8X,'AIRWAYS',7X,'Q(M3/SEC)',4X,'X(M)',3X,'A(M2)'/)
       READ(7,24) ITER,IPR
  24   FORMAT(20X,2I10)
         IF(ITER.LT.IPR)MPRINT=1
C-----------------------------------------------------------------------
C                READ TUNNELS' AIRFLOW AND DIMENSIONS
C-----------------------------------------------------------------------
          N=0
   15     N=N+1
   16     READ(7,20)IRAD(1,N),IRAD(2,N),RAD(1,N),RAD(2,N),RAD(3,N)
   20     FORMAT(2I5,20X,F10.5,20X,2F10.5)
C  20     FORMAT(2(I6,2X),3F8.1)
          IF(IRAD(2,N).EQ.1)GO TO 16
          IF(IRAD(1,N).LT.1) GO TO 25
C-----------------------------------------------------------------------
C                    REVERSE NODES IF FLOW NEGATIVE
C-----------------------------------------------------------------------
          IF(RAD(1,N).GE.0.0)GO TO 21
             RAD(1,N)=-RAD(1,N)
             TN=IRAD(1,N)
             IRAD(1,N)=IRAD(2,N)
             IRAD(2,N)=TN
   21     WRITE(6,22)IRAD(1,N),IRAD(2,N),RAD(1,N),RAD(2,N),RAD(3,N)
   22     FORMAT(2I8,6X,F8.1,1X,2F8.1)
          GO TO 15
C-----------------------------------------------------------------------
C                          NA = NO. OF AIRWAYS
C-----------------------------------------------------------------------
   25     NA=N-1
C-----------------------------------------------------------------------
C            READ DATA FOR NODES WITH KNOWN INPUT PARAMETERS
C                                     AND FILTRATION UNITS
C-----------------------------------------------------------------------
       N=0
   27     N=N+1
          READ(8,30)NPAR(N),PAR(1,N),PAR(2,N)
   30     FORMAT(I10,2F10.3)
           IF(NPAR(N).LT.1)GO TO 32
          GO TO 27
C-----------------------------------------------------------------------
C              NP = NO. OF NODES WITH KNOWN INPUT PARAMETERS
C-----------------------------------------------------------------------
   32     NP=N-1
C-----------------------------------------------------------------------
C                          READ EMANATION DATA
```

```
C------------------------------------------------------------
      WRITE(6, 34)
   34 FORMAT(////8X, 'AIRWAYS', 8X, 'AREA', 4X, 'EM RATE')
      WRITE(6, 36)
   36 FORMAT(23X, '(M2)', 4X, '(BQ/(M2.SEC))'//)
   40  I=0
       NT=0
       READ(8, 42)NB, NE
   42  FORMAT(2(I6, 2X))
       IF(NB.LT.1) GO TO 60
C------------------------------------------------------------
C                    NB = BEGINNING NODE
C                    NE = END NODE
C------------------------------------------------------------
       READ(8, 45) NK
C------------------------------------------------------------
C             NK = NO.  OF DIFFERENT EMANATING COMPONENTS
C------------------------------------------------------------
   45  FORMAT(I4)
   43  I=I+1
       IF(I.LE.NA)GO TO 44
       IF(NT.EQ.0)GO TO 46
C------------------------------------------------------------
C              DUMMY READ IF BRANCH NOT IN CURRENT NETWORK
C------------------------------------------------------------
       READ(8, 50)(EM(1, K, I), EM(2, K, I), K=1, NK)
       GO TO 40
C------------------------------------------------------------
C                     CHECK FOR REVERSED BRANCH
C------------------------------------------------------------
   46  I=1
       NT=NB
       NB=NE
       NE=NT
   44  IF(IRAD(1, I).NE.NB) GO TO 43
       IF(IRAD(2, I).NE.NE) GO TO 43
       WRITE(6, 47) NB, NE
   47  FORMAT(2I8)
C------------------------------------------------------------
C                   READ EMANATING CHARACTERISTICS
C------------------------------------------------------------
       READ(8, 50)(EM(1, K, I), EM(2, K, I), K=1, NK)
   50  FORMAT(F8.1, F8.3)
       WRITE(6, 52)(EM(1, K, I), EM(2, K, I), K=1, NK)
   52  FORMAT(20X, F8.1, F8.3)
       GO TO 40
   60  CONTINUE
C------------------------------------------------------------
C              CALCULATE WORKING LEVEL GROWTH IN AIRWAYS
C              HAVING KNOWN INPUT PARAMETERS
C------------------------------------------------------------
      DO 70 N=1, NP
        IK=0
        IF(PAR(1, N).GE.0)GO TO 62
        WRITE(6, 65)NPAR(N)
   65   FORMAT(/10X, 'FILTRATION UNIT AT NODE ', I5)
        GO TO 70
C------------------------------------------------------------
```

```
C              DETERMINE AIRWAYS STARTING FROM KNOWN NODE
C-----------------------------------------------------------------
   62   IK=IK+1
        IF(IK.GT.NA)GO TO 70
        IF(NPAR(N).NE.IRAD(1,IK)) GO TO 62
C-----------------------------------------------------------------
C                   INITIALIZE TIME, WLO AND CO
C-----------------------------------------------------------------
        RAD(4,IK)=0.0
        RAD(5,IK)=PAR(1,N)
        RAD(6,IK)=PAR(2,N)
        CALL WORKL(IK,IRAD,RAD,EM)
        IRAD(3,IK)=-1
        I=IK
        GO TO 62
   70 CONTINUE
C-----------------------------------------------------------------
C                   I ==> LAST CALCULATED AIRWAY
C-----------------------------------------------------------------
   74 IFLAG=0
   75 K=0
      M=0
C-----------------------------------------------------------------
C                  JE = END NODE OF PREVIOUS AIRWAY
C-----------------------------------------------------------------
        JE=IRAD(2,I)
C-----------------------------------------------------------------
C            IFLAG = COUNTER TO FIND RECIRCULATION LOOPS
C            LFLAG = DEFINES TYPE OF LOOP
C-----------------------------------------------------------------
        IF(LFLAG.GT.0)GO TO 80
        IFLAG=IFLAG+1
        ISCAN(IFLAG)=JE
        IF(IFLAG.LT.3)GO TO 80
C-----------------------------------------------------------------
C                  CHECK FOR RECIRCULATION LOOPS
C-----------------------------------------------------------------
        DO 79 KK=2,IFLAG
          IF(ISCAN(KK-1).NE.JE)GO TO 79
          NM=0
          LLOOP=0
          LOOP=IFLAG-KK+1
          DO 77 J=1,LOOP
            JJ=IFLAG+1-J
C-----------------------------------------------------------------
C                    FIND BRANCHES IN LOOP
C-----------------------------------------------------------------
            JA=ISCAN(JJ)
            JB=ISCAN(JJ-1)
            NL=0
            DO 76 L=1,NA
              IF(IRAD(1,L).EQ.JA.AND.IRAD(2,L).EQ.JB)JL=L
              IF(IRAD(2,L).NE.JA)GO TO 76
              IF(IRAD(3,L).LT.0)NL=NL+1
C-----------------------------------------------------------------
C                     CHECK FOR COMPOUND LOOPS
C-----------------------------------------------------------------
              IF(IRAD(3,L).GE.0)NL=NL+10
```

```
  76        CONTINUE
C------------------------------------------------------------------
C                  SELECT BRANCH TO START ITERATIONS
C------------------------------------------------------------------
            ILOOP(J)=JL
            IF(NM.GE.NL)GO TO 77
            NM=NL
            IL=JL
  77      CONTINUE
          LFLAG=NM/10+1
          IF(LFLAG.EQ.2)WRITE(6,12)(ISCAN(LL),LL=KK,IFLAG)
  12      FORMAT(//5X,'** SIMPLE RECIRCULATION LOOP: '/12I4)
          IF(LFLAG.GT.2)WRITE(6,13)(ISCAN(LL),LL=KK,IFLAG)
  13      FORMAT(//5X,'** COMPOUND RECIRCULATION LOOP: '/12I4)
          IF(LFLAG.GT.2)ILL=IL
          CALL WORKL(IL,IRAD,RAD,EM)
          IRAD(3,IL)=1
          RAD(11,L)=RAD(8,IL)
          I=IL
          GO TO 74
  79    CONTINUE
C------------------------------------------------------------------
C                   DETERMINE AIRWAYS LEADING TO JE
C------------------------------------------------------------------
  80 K=K+1
      IF(K.GT.NA)GO TO 90
      IF(IRAD(2,K).NE.JE)GO TO 80
C------------------------------------------------------------------
C                 M = NO. OF AIRWAYS WITH SAME END NODE
C------------------------------------------------------------------
      M=M+1
      ICOMB(M)=K
C------------------------------------------------------------------
C                     HAS AIRWAY BEEN DEALT WITH
C------------------------------------------------------------------
      IF(LFLAG-2)82,81,80
  81 IF(IRAD(3,K).GT.0)GO TO 80
  82 IF(IRAD(3,K).LT.0)GO TO 80
      I=0
  85 I=I+1
C------------------------------------------------------------------
C           DETERMINE AIRWAYS LEADING TO UNCALCULATED AIRWAY
C------------------------------------------------------------------
      IF(IRAD(2,I).EQ.IRAD(1,K)) GO TO 75
      GO TO 85
  90 CONTINUE
C------------------------------------------------------------------
C              DETERMINE AVERAGE AGE, WLO AND CO VALUES
C              WITH RESPECT TO FLOW RATES
C------------------------------------------------------------------
        CO=0.
        W=0.
        T=0.
        FLOW=0.
        LOK=1
        DO 120 K3=1,M
          K2=ICOMB(K3)
          IF(IRAD(3,K2).GE.0)LOK=0
```

```
            T=T+RAD(4,K2)*RAD(1,K2)
            W=W+RAD(7,K2)*RAD(1,K2)
            CO=CO+RAD(8,K2)*RAD(1,K2)
            FLOW=FLOW+RAD(1,K2)
  120     CONTINUE
C-----------------------------------------------------------------------
C                       CHECK FOR FILTRATION UNIT
C-----------------------------------------------------------------------
          DO 100 N=1,NP
            IF(NPAR(N).NE.JE)GO TO 100
            IF(PAR(1,N).LT.0.0)W=W*(1-PAR(2,N)/100.0)
   100    CONTINUE
          I=0
C-----------------------------------------------------------------------
C         CANCEL RECIRCULATION LOOP PROCESSING IF NODE INPUT KNOWN
C-----------------------------------------------------------------------
          IF(LOK*LFLAG.EQ.0)GO TO 104
          LFLAG=0
          KLOOP=0
          WRITE(6,152)
   152    FORMAT(/10X,'**LOOP CANCELED**')
   104    L=0
C-----------------------------------------------------------------------
C                       FIND ALL BRANCHES OUT OF NODE
C-----------------------------------------------------------------------
   105    L=L+1
          IF(L.GT.NA)GO TO 130
          IF(IRAD(1,L).NE.JE)GO TO 105
          IF(IRAD(3,L).LT.0)GO TO 105
          IF(LFLAG.NE.2)GO TO 110
C-----------------------------------------------------------------------
C                   PROCESS LOOP BRANCH ONLY IF SIMPLE LOOP
C-----------------------------------------------------------------------
           DO 108 J=1,LOOP
           IF(L.EQ.ILOOP(J))GO TO 110
   108     CONTINUE
          GO TO 105
C-----------------------------------------------------------------------
C                      TRANSFER VALUES TO NEW BRANCH
C-----------------------------------------------------------------------
  110 RAD(4,L)=T/FLOW
        RAD(5,L)=W/FLOW
        RAD(6,L)=CO/FLOW
        CALL WORKL(L,IRAD,RAD,EM)
          IRAD(3,L)=LFLAG-LOK
          I=L
          IF(LFLAG.NE.2)GO TO 105
          IF(L.NE.IL)GO TO 75
C-----------------------------------------------------------------------
C                       CHECK FOR STEADY STATE IN LOOP
C-----------------------------------------------------------------------
          LLOOP=LLGOP+1
          IF(ABS(RAD(11,L)-RAD(8,L)).LE.XXX)GO TO 112
          RAD(11,L)=RAD(8,L)
          GO TO 75
C-----------------------------------------------------------------------
C                     SET LOGIC SWITCHES FOR SOLVED LOOP
C-----------------------------------------------------------------------
```

```
  112   DO 114 J=1, LOOP
          LJ=ILOOP(J)
          IRAD(3, LJ)=-1
  114   CONTINUE
        WRITE(6, 155)LLOOP
        LFLAG=0
        LOK=1
        L=0
        GO TO 105
C-------------------------------------------------------------------
C                      ALL AIRWAYS DEALT WITH
C                      IF NOT, SELECT NEXT AIRWAY
C-------------------------------------------------------------------
  130 IF(LFLAG. GT. 2)I=0
      DO 135 L=1, NA
        IF(IRAD(3, L). NE. 0)GO TO 135
        IF(I. NE. 0)GO TO 74
        DO 132 I=1, NA
          IF(IRAD(2, I). EQ. IRAD(1, L))GO TO 74
  132   CONTINUE
  135 CONTINUE
      ITEST=0
      DO 200 L=1, NA
        IF(IRAD(3, L). LT. 0)GO TO 200
C-------------------------------------------------------------------
C        TEST FOR STEADY STATE IN ALL BRANCHES OF COMPOUND LOOP
C-------------------------------------------------------------------
        IF(ABS(RAD(11, L)-RAD(8, L)). GT. XXX)ITEST=1
        RAD(11, L)=RAD(8, L)
        IRAD(3, L)=0
  200 CONTINUE
      KLOOP=KLOOP+1
C-------------------------------------------------------------------
C                 TERMINATE IF ITERATION LIMIT EXCEEDED
C-------------------------------------------------------------------
      IF(KLOOP. GT. ITER)GO TO 150
       IF(KLOOP. GT. ITER-IPR)MPRINT=1
      IF(ITEST. EQ. 0)GO TO 150
        I=ILL
        GO TO 75
C-------------------------------------------------------------------
C                          PRINT RESULTS
C-------------------------------------------------------------------
  150 IF(LFLAG. GT. 2)WRITE(6, 155)KLOOP
  155 FORMAT(6X, 'NO. OF ITERATIONS FOR LOOP ', I5/)
      WRITE(6, 140)
  140 FORMAT('1'////1X, 'INITIAL', 1X, 'FINAL', 1X, 'WORKING', 1X, 'RADON',
     *2X, 'BRANCH', 3X, 'EQUILIBRIUM', 2X, 'RADIOACTIVE')
      WRITE(6, 141)
  141 FORMAT(2X, 'NODE', 3X, 'NODE', 3X, 'LEVEL', 2X, 'CONC. ', 2X, 'TRAVEL',
     *5X, 'RATIO', 7X, 'AGE(min)')
      WRITE(6, 143)
  143 FORMAT(22X, '(BQ/M3)'1X, 'TIME(min)')
      DO 160 I=1, NA
      WRITE(6, 145) IRAD(1, I), IRAD(2, I), RAD(7, I), RAD(8, I), RAD(9, I),
     +RAD(10, I), RAD(4, I)
  145 FORMAT(I6, 1X, I5, 1X, F7. 4, 2X, F7. 2, 2X, F7. 3, 4X, F7. 3, 3X, F7. 3)
  160 CONTINUE
```

```
C
      STOP
      END
C
C***********************************************************************
C
      SUBROUTINE WORKL(L)
C
C             ============================================
C                THIS SUBROUTINE CALCULATES THE GROWTH OF
C               RADON-RELATED PARAMETERS THROUGH AN AIRWAY
C             ============================================
C
C-----------------------------------------------------------------------
C   INPUT ARGUMENT  L = AIRWAY NUMBER
C   --------------
C-----------------------------------------------------------------------
      COMMON IRAD(3,70),MPRINT,RAD(11,70),EM(2,5,70)
C-----------------------------------------------------------------------
C                 CALCULATE TIME OF RESIDENCE, T, IN MINUTES
C                           VOL = TUNNEL VOLUME
C-----------------------------------------------------------------------
      VOL=RAD(2,L)*RAD(3,L)
      T=VOL/(RAD(1,L)*60.0)
C-----------------------------------------------------------------------
C                    DETERMINE VOLUMETRIC AGE IN MINUTES
C-----------------------------------------------------------------------
      RAD(9,L)=T
      WL1=0.0
      IF(RAD(6,L).LT.0.01)GO TO 19
C-----------------------------------------------------------------------
C                      ER = INITIAL EQUILIBRIUM RATIO
C-----------------------------------------------------------------------
      ER=RAD(5,L)*3700./RAD(6,L)
C-----------------------------------------------------------------------
C                       TR = RADIOACTIVE AGE OF AIR
C-----------------------------------------------------------------------
      IF (ER.LT.0.001) GO TO 19
      IF (ER.GT..9876)                 TR=220.
      IF (ER.GT..9088 .AND.ER.LE..9876)TR=(ER/0.5076000)**8.1033
      IF (ER.GT..55234.AND.ER.LE..9088)TR=(ER/0.0856601)**1.9984
      IF (ER.LE..55234)                TR=(ER/0.0227672)**1.1680
      TT = TR + T
C-----------------------------------------------------------------------
C                  A = EQUILIBRIUM RATIO AT END OF AIRWAY
C-----------------------------------------------------------------------
      IF (TT.GE.220.)                   A=1.0
      IF (TT.GE.112.0 .AND.TT.LT.220.0) A=0.5076    *TT**0.123405
      IF (TT.GE.41.45 .AND.TT.LT.112.0) A=0.0856601*TT**0.5004
      IF (TT.LT.41.45)                  A=0.0227672*TT**0.856139
      WL1=A*RAD(6,L)/3700.
C-----------------------------------------------------------------------
C                      ES = TOTAL RADON EMANATION
C-----------------------------------------------------------------------
   19 ES=0.0
      WL2=0.0
      EV=0.0
      IF(VOL.LE.0.0)GO TO 26
      DO 20 K=1,5
        ES=ES+EM(1,K,L)*EM(2,K,L)
   20 CONTINUE
C-----------------------------------------------------------------------
C                         EV = ES / TUNNEL VOLUME
C-----------------------------------------------------------------------
      EV=ES/VOL
C-----------------------------------------------------------------------
C                     WL2 = WL GROWTH FROM DIFFUSED RADON
C-----------------------------------------------------------------------
      IF (T.GT.220.)                WL2=EV*(.0162*T-.8486)
      IF (T.LE.220.AND.T.GT.112.) WL2=EV*(7.328E-3*T**1.123405-.41728
      IF (T.LE.112.AND.T.GT.41.45)WL2=EV*(9.258E-4*T**1.500400-.04743
      IF (T .LE.41.45)             WL2=EV* 1.989E-4*T**1.856139
C-----------------------------------------------------------------------
C                                SUMMATE FOR TOTAL WL
C-----------------------------------------------------------------------
   26 RAD(7,L)=WL1+WL2
      CA=EV*T*60.0
C-----------------------------------------------------------------------
C                          CA = DIFFUSED RADON ADDED
C                      CALCULATE CONCENTRATION REMAINING
C-----------------------------------------------------------------------
      RAD(8,L)=RAD(6,L)+CA
       IF(MPRINT.EQ.1)WRITE(6,1000)(IRAD(I,L),I=1,3),RAD(4,L),
     *RAD(5,L),RAD(6,L),RAD(7,L),RAD(8,L),RAD(9,L)
 1000 FORMAT(3I5,6F10.4)
C-----------------------------------------------------------------------
C                      CALCULATE FINAL EQUILIBRIUM RATIO
C-----------------------------------------------------------------------
      IF (RAD(8,L).EQ.0.) RETURN
      RAD(10,L)=RAD(7,L)*370000.0/RAD(8,L)
      ER=RAD(10,L)/100
      IF(ER.LT.0.001)  RAD(4,L)=0
      IF(ER.GT..9876)  RAD(4,L)=220
      IF(ER.GT..9088.AND.ER.LE..9876) RAD(4,L)=(ER/0.5076)**8.1033
      IF(ER.GT..55234.AND.ER.LE..9088)RAD(4,L)=(ER/0.08566)**1.9984
      IF(ER.LE..55234)RAD(4,L)=(ER/0.0227672)**1.1680
C
      RETURN
      END
```

APPENDIX E

REVIEW OF EXISTING MINE VENTILATION MODELS

Table of Contents

Page

E.1 Introduction E-1

E.2 Discussion of Model Concepts E-1
  E.2.1 Incompressible Flow Programs E-2
  E.2.2 Compressible Flow Programs E-13

E.3 Discussion of Major Existing Models E-15

E.4 Practical Problems in Model Application E-26
  E.4.1 Model Limitations E-26
  E.4.2 Other Factors Affecting Model Predictions of a Mine Ventilation System E-30

References

REVIEW OF EXISTING MINE VENTILATION MODELS

E.1 Introduction

The first task of this study involved the review of available literature on mine ventilation and existing ventilation models. This was initiated by studying the large amount of literature already available to the study team and by contacting persons and institutions familiar with mine ventilation practical designs and ventilation models.

A number of fan manufacturers in the Toronto area were also contacted to discuss their experience in representing fan characteristics in computer simulations.

In addition to the above, on-line computer literature searches were performed to ensure complete coverage of published materials. Data bases were assessed via the U.S.-based DIALOG system and the Canadian-based QL Systems Limited.

What follows is an outline of the various concepts used in mine ventilation modelling presented in terms of input and output structure. The subsequent sections discuss the major existing mine ventilation models and highlight their features and potential inadequacies.

E.2 Discussion of Model Concepts

The most commonest application of computers to mine ventilation is in the Hardy-Cross network analysis procedure. In the simplest implementation, the airflow is treated as a constant density medium. A network of nodes and branches is established to simulate the airflow paths through the mine and each branch is assigned a resistance value. A pressure source or fixed flow is specified in one or more of the branches and the solution yields the distribution of airflow through all branches of the network.

Several programs of this type are in use with varying degrees of enhancements to the basic input and output structure. Some of the features available, with comments on their utility, are discussed below. For a general discussion of the various fundamental concepts of mine ventilation, the reader is referred to Burrows et al (1982) and Hartmann (1982). It should be realized that the fundamental objective of all mine ventilation models is the calculation of air quantity at any point in the mine.

E.2.1 Incompressible Flow Programs

The term "incompressible" does not allow for changes in air density due to changes in pressure. However, changes in pressure due to humidity or temperature changes can be accounted for under "incompressible" flow.

E.2.1.1 Input Structure

Branch Resistances

The resistance values assigned to the individual branches of the circuit govern the accuracy of the simulation. In practice, these values may be difficult to determine accurately. For an existing mine they may be calculated from measurements of pressure drop and flow rate through a given airway. However, there are great practical difficulties in measuring these parameters accurately, particularly at low flow rates and in short airways.

Another technique is to calculate resistance from airway geometry and an estimated airway friction factor (K factor). The K factor appearing in the widely-used Atkinson equation is dependent on the units of pressure and flow employed and one must be careful when adopting published values to ensure that they are in the right units. A more satisfactory approach is to use the more fundamental hydraulic formula with its dimensionless friction factor which can be calculated directly from the airway dimensions and a representative surface roughness value. This latter method offers the best flexibility in terms of ability to model the effects of changes in airway size.

In this case, the airway dimensions form the input data and the branch resistances are calculated by the program. Using S.I. units, the appropriate equation is:

$$\text{Airway resistance } R\ (\text{kg/m}^7) = pfLC/8A^3 \qquad \text{(E.1)}$$

where

p = air density (kg/m^3)
f = airway friction factor
L = airway length (m)
C = perimeter of airway cross-section (m)
A = cross-sectional area of airway (m^2)

The parameter group $[pf/8]$ takes the place of the K factor in the Atkinson equation.

Since airflow in most underground airways is well into the turbulent range (high Reynolds Number), the friction fraction f is dependent only on the characteristics of the surface and is given by:

$$f = 1/[2 \log(14.8A/Ce)]^2 \qquad \text{(E.2)}$$

where

e = roughness height (m)

Shock losses due to bends or changes in area can also be added where appropriate, expressed either as separate resistance values or as additional equivalent lengths of airway.

Fixed Flow Branches

The ability to specify a constant flow rate in a designated branch is a very powerful and useful feature. It is used in preliminary design runs to determine the required fan duty points and the location and type of controls needed to achieve the desired airflow distribution.

Regulators

Control of airflow distribution in a mine is achieved by booster fans and regulators. A regulator is a variable resistance placed in a branch to restrict the flow to the desired value. For simulation purposes, a maximum flow value is specified for a regulator branch.

Laminar Flow Branches

Airflow through caved ground, or an orepass full of broken muck, or leakage through ventilation doors can be more realistically modeled by a branch with a linear characteristic (i.e. $P = RQ$). In this case, the resistance value would have to be determined by measurement.

Fans

The simplest way to input a fan characteristic is to enter a series of pressure and volume coordinates from the useful operating range of the fan curve. Typically, the program interpolates linearly between the points. Some programs fit a polynomial to the points in an attempt to match the fan curve more closely. The extra computation involved in this technique is often not warranted due to the imprecise nature of most fan curves. Many fans have different blade settings to allow them to operate efficiently over a broad range of duty. For a desired system operating point, the points of the input fan characteristic located closest to the desired operating point are selected by computer. The solution gives the required fan inlet damper blade setting (centrifugal fans) or the required blade angle (axial fans).

When new fans have to be selected, a selection is made by choosing a fan speed and calculating the corresponding specific fan speed. In this case the computer can contain curves giving optimum fan efficiency and specific diameter. The selection is made by printing out the best fan designation for the fan diameter found. Fans should, under best operating conditions, be 82 to 87% efficient. In most programs, separate curves must be input for each blade setting, and the appropriate curve must be assigned for a particular run of the program. It should be feasible to include a range of settings with each fan curve and have the program select the appropriate setting or print a message if the fan is unsuitable.

There is also variability in the method of associating fans with branches of the circuit. The best method is that in which the fan data are maintained in a separate file and associated with the desired branch by a serial number. In this way, each fan has only to be defined once, and multiple fan installations can be simulated by inserting additional fan branches, in series or in parallel as appropriate thus avoiding the explicit specification of combined fan characteristics.

Alternatively, if the number and arrangement of the fans is specified, the program can calculate the combined characteristics and assign the resulting fan curve to a single branch. This saves on the number of branches and improves the efficiency of the network solution.

When digitizing fan curves, one must be aware of the various conventions for expressing fan pressuras and efficiencies. Formerly, the practice of fan manufacturers was to publish fan data in the form of fan static pressure characteristics, since only the fan static pressure is useful in overcoming mine resistance. Data from these curves can be applied directly to the mine circuit.

However, some manufacturers have adopted the practice of publishing their fan data in the form of fan total pressure characteristics. (Total pressure is the fan static pressure plus velocity pressure at the fan outlet.) In this case, an extra resistance should be added to each mine exhaust airway to account for the velocity pressure (kinetic energy) loss at exit. Since velocity pressure is given by $P = pV^2/2$ and since $R = P/Q^2$, the extra resistance is:

$$R = pV^2/2Q^2 \qquad \text{(E.3a)}$$

where

V = air velocity (m/s)
Q = volumetric flow (m^3/s)

and the other terms are as previously defined. Substituting Q=VA, this reduces to

$$R = p/2A^2$$
$$= 0.6/A^2 \qquad \text{(E.3b)}$$

for resistance at standard density, where A is the area of the mine airway at the point where the air leaves the mine. (This formula also affords a convenient way of assessing the effect of an evasee exit on an exhaust fan.)

Natural Ventilation Pressure

In a simple incompressible flow model, the effect of natural ventilation pressure (NVP) is usually incorporated by adding a constant pressure value (positive or negative) to appropriate branches in the system (usually the shafts). The values used must be calculated separately and are normally based on known or estimated temperature and elevation differences in the mine. There are a number of methods of calculating the NVP and this is a rather controversial topic. Values change with the seasons. The difficulty of assigning realistic values of NVP is one of the chief shortcomings of the incompressible flow model, particularly in deeper mines.

Gas Inflows

Some programs, particularly those developed for use in coal mines, have provision for gas inflows. The effect of compressed air use can be easily modeled by introducing a fixed-flow branch from surface to the point of liberation of the compressed air.

In other mines compressed air use is variable since it is only used at active working places during the drilling shifts. The release is minimal at other times. A fixed flow branch from surface could be used to model these situations.

Units

With an incompressible flow program, it is possible to write the program in such a way that any consistent system of units may be used by the operator and the output interpreted accordingly. In this case, the airway resistance values would have to be pre-calculated and entered directly since they are unit dependent. If the program is required to calculate resistances from airway dimensions, an appropriate factor would have to be entered depending on the units chosen for pressure and flow. Another factor would be required to relate the product of pressure and flow to power units.

Maximum Number of Iterations and Precision

The Hardy-Cross algorithm is an iterative technique which starts with an initial estimated flow distribution and gradually adjusts the branch flows until the pressure drops through the circuit are fully balanced.

Adjustments to the flow values normally diminish rapidly with each iteration, and when the largest flow correction in any mesh is less than the degree of precision specified, a solution has been reached.

Although the method is fairly robust, it is possible to have an unstable circuit in which the flow in one or more branches may fluctuate about zero. This is more likely to happen in complex circuits with multiple fans and several openings to surface. Sometimes, an incorrect fan specification might also cause instability. It is therefore necessary to specify a limit to the number of iterations to prevent the program from running indefinitely.

E.2.1.2 Output Structure

Input Data Listing

A listing of the input branch data and fan data is generally included to facilitate checking of the data. This output should be optional, as once the basic circuit is established, many runs may be made with only minor changes and only the changes need to be listed on the printout.

Airflow Distribution

The quantity and direction of airflow in each branch is the standard output produced by any mine ventilation program. Values are normally listed for the whole circuit with the branches identified by the number of end nodes. The direction of flow is implied by the order of the nodes. (i.e., flow is from the first node to the second node. If flow has a negative sign, flow is in the reverse direction.) The airflow values form part of the input data requirement for the radiation program, so for the purposes of this project, they should also be retained in a disk file after each run.

Network Plotting

For a large circuit it can be very difficult and time-consuming to evaluate the airflow distribution from a computer listing. A schematic plot of the network with flow values and directions appropriately represented can be a valuable aid to rapid assimilation of the system. A plotting option requires additional input information in the form of a file of node coordinates. It also requires a plotting device.

Effects of Changes to the Circuit

Another means of facilitating analysis of the effects of minor changes to the circuit is to have a listing of branches showing a change in flow of more than 10% (or some other criterion specified by the user). Because of imprecision in the input data and the hour-to-hour changes in a dynamic mine ventilation system, changes of less than 10% have no practical significance.

Pressure Drop

The pressure drop in each branch due to airway resistance is generally included as part of the standard branch data output.

Power Absorbed

The pressure drop in an airway multiplied by the flow rate through the airway, when multiplied by a suitable conversion factor depending on the units used, gives the power (horsepower or kW) absorbed by the branch. This parameter is also normally included as part of the standard branch data output. Since power use can be directly related to cost, this affords a convenient way to identify high operating cost parts of the circuit.

Fixed Flow Branches

Where a fixed flow value is specified in the input, the output should indicate what resistance is required to achieve that flow rate. A message can then be printed indicating whether a regulator is required to increase the branch resistance, or a fan is required to effectively reduce the resistance. In the latter case, the required fan operating point is indicated for fan selection purposes.

Regulator Branches

Output for a regulator branch indicates whether indeed a regulator is needed to restrict the flow to the value specified and, if so, the additional resistance necessary. As an extension, the program could also calculate the orifice size necessary to generate this resistance.

Note that although the regulator branch specifies a maximum flow value, there is effectively no limit to flow in the opposite direction if the network solution determines that the airflow in the branch actually flows in the opposite direction to that implied by the order of the nodes. The program could be coded in such a way that the regulator works in both directions but the "UNRESTRICTED FLOW" message precipitated by a reverse flow situation serves to draw attention to a probable error or undesirable condition in the circuit.

If it is desired to limit flow in the opposite direction to that implied by the node order of a regulator branch, it is merely necessary to specify the maximum flow value with a negative sign.

Fan Branches

Output for a fan branch gives the flow rate and pressure rise through the fan. Normally, the power is also calculated. The product of flow volume and pressure gives the air power or power output by the fan. If the efficiency curve is also included in the fan characteristic input data, the input power can be calculated and, if a unit power cost is also input, the operating cost of the fan can be computed on an hourly or annual basis.

If the fan is unsuitable or is operating in a stall mode or outside its recommended performance range, appropriate messages can be printed drawing attention to the condition. Although these conditions would require re-running of the program and a trial and error method of fan adjustment, this is considered preferable to the practice of eliminating a stalled fan from the circuit entirely and automatically re-running the program without it, as is done in some programs.

Cost

Some programs print the power cost incurred by each branch by simply applying a factor to the power absorbed figures. While the use of a dollar value may emphasize the high cost branches, the costs only reflect the air power, or, at best, only include an arbitrary efficiency factor. It is considered more useful to just list the power absorbed in each branch and assign costs to the fans as outlined above. This gives a truer cost picture as it includes the individual fan efficiency and discounts any "free" energy provided by natural ventilation pressure.

Node Pressures

The pressure at each node, relative to atmospheric pressure, can be readily calculated for the final balanced solution. This output is useful as an option, particularly where a push-pull system is used.

This output can be used to determine pressure drops across doors or stoppings, information which is useful for circuit validation tests and planning exercises.

Airway Velocities

If the airway dimensions form part of the input data, the branch flow rates can easily be translated to velocities. These values can be included with the standard branch output listing or included as an optional additional output. Air velocity is an important ventilation parameter from the point of view of cooling effect, dust pick-up and the miner's perception that ventilation is, in fact, being provided. Air velocity is also a requisite parameter in calculating the "age" of the air in radiation models.

Mesh Selection

A printout of the node connection order of the meshes selected by the program is useful in certain circumstances. For instance, when first validating a mine circuit, it is sometimes necessary to make adjustments to some of the branch resistance values to make the computer solution agree with measured data from the mine. This is often a trial and error process but examination of the meshes can give some clues to the branches which should be changed. Identification of the meshes is also useful when assigning natural ventilation pressures.

E.2.2 Compressible Flow Programs

Sometimes referred to as thermodynamic programs, these programs take into account the variations in density of mine air which are principally brought about by changes in elevation, temperature and moisture content. The full thermodynamic treatment is quite complex and requires a greater amount of input data. In addition to the airway resistance and shock loss data required for the hydraulic model, the thermodynamic program requires the elevation of each node, the surface barometric pressure and wet-and-dry bulb temperatures, temperature gradient and heat transfer characteristics of the wall rocks, and the location and magnitude of any sources of heat and moisture in the mine. As an alternative to the heat and moisture data, the wet-and-dry bulb temperatures and barometric pressure at each node of the circuit may be input. In practical terms, these data are hard to come by and we must ask ourselves whether the precision of the mathematical treatment is negated by the approximate and ephemeral nature of the available data.

From a practical point of view, a satisfactory compromise seems to be offered by a "pseudo-thermodynamic" approach suggested by Greuer (1979) and further elaborated by Stachulak (1979). These authors show, by manipulation of the fundamental thermodynamic equations for airflow in a closed mesh in a mine ventilation circuit, that the effects of density change due to elevation and temperature may be fully accounted for by a natural ventilation pressure (NVP) term in each mesh. They also offer a simple means to calculate the natural ventilation pressure.

For each mesh, the natural ventilation head (expressed as a column of air) is given by

$$\text{natural ventilation head} = \Sigma(T\,\Delta Z)/T_m \qquad \text{(E.4)}$$

where

T = the average absolute temperature in a branch

Z = the difference in elevation between the ends of the branch

T_m = the average absolute temperature in the mesh.

The summation term represents the sum of the temperature elevation products for each branch in the mesh.

This scheme is very easily added to a standard incompressible flow program. After the mesh selection process, the NVP is calculated and assigned to the independent branch for each mesh.

This method also requires specification of elevation and temperature of each node in the network. However, if such information is not available, the circuit is simply solved as an incompressible flow network without the NVP values.

The effects of moisture are not included in this procedure. Further study is required to determine under what conditions of temperature and depth the moisture changes become significant compared to the effects of other areas of uncertainty in the data.

E.3 Discussion of Major Existing Models

Penn State University Mine Ventilation Simulator (PSU/MVS)

This is one of the better known incompressible flow programs (Ramani 1975). It has been developed from the earlier Wang and Saperstein (1970) program and supports many extra features including the following:

- Alternative means of entering branch resistance data.
- Extensive error testing and reporting.
- Polynomial curve fitting for fan data.
- Simulation of gas inflows.
- Fixed quantity branches.
- Limited flow branches.
- Node pressure output.
- Calculation of gas concentration in each branch (methane).
- Leakage through stoppings.
- Assigned values of NVP.

- Extensive output report.
- English units only.

The package was designed primarily for the U.S. coal industry, development being sponsored by the U.S. Bureau of Mines. (U.S.B.M.) It has fairly sophisticated routines for simulating fan curves, changing fan speeds, and combining fans in various series and parallel combinations.

Mining engineering students at Queen's University have found this program cumbersome to use, with respect to both entering the data and assimilating the calculated results.

Michigan Tech Mine Ventilation Program

This program was originated in Germany by R.E. Greuer in the 1960's and was further developed under sponsorship of the U.S.B.M. to model the effect of mine fires on the mine ventilation circuit (Greuer 1979; Stachulak 1979). It is able to model methane emissions as well as smoke production from mine fires.

It is a pseudo-thermodynamic program which accounts for elevation and temperature effects by calculating a natural ventilation term for each mesh of a network. It has been used by INCO to model the ventilation system at the 2500 m deep Creighton mine where there are appreciable thermal effects (Stachulak 1979). INCO are currently investigating alternative natural ventilation programs which can be implemented in a microcomputer system (Coulter 1984).

In order to calculate the NVP values, elevation and temperature data must be input for each node. The program uses English units and is poorly structured, consisting of one long main program with no subroutines and minimal comments. A copy available in the Queen's Mining Department has been converted to S.I. units. The specification of fan data is handled inefficiently.

Queen's Mining Ventilation Program VPROG

This is a constant density program, the nucleus of which was written by P. Muzard of Chile. It has been used as a teaching tool for many years and has been used to study the ventilation systems at several Canadian mines. It is freely available to Queen's mining students and copies have been distributed to various mining companies and consulting firms. Several improvements have been made to the original program, the latest being the incorporation by Greuer of the natural ventilation pressure scheme to upgrade the program to pseudo-thermodynamic status.

Other features supported by the program are as follows:

- Optional suppression of input data listing.
- Network plotting.
- Ability to make temporary changes to the circuit without affecting the basic network.
- Fixed flow branches.
- Limit flow (regulator) branches.
- Laminar flow branches.
- Accepts any consistent system of units.
- Optional creation of new branch file from any balanced solution.
- Node pressures relative to atmospheric.
- Separate output listing of branches showing change in flow of 10% or more from previous solution.

Other useful features which could be added to the program include:

- Fan efficiency characteristics and unit power cost to allow computation of fan operating costs.
- Air velocity in each branch. This will require input of the airway dimensions which are currently not used since resistance values are entered directly. At this point it will probably become too unwieldy to maintain unit independence and separate versions of the program will be required for the different unit systems.
- Means of handling the effects of changes in moisture content.

Colorado School of Mines Ventilation Program

This is a full thermodynamic treatment requiring the data from a mine resistance pressure survey for input (Morse 1977). At each end of each branch, the parameters required are:

- Barometric pressure
- Dry-bulb temperature
- Wet-bulb temperature
- Elevation
- Volumetric airflow

Also required are the branch dimensions, virgin rock temperature, wall rock heat transfer coefficient, heat sources, and moisture additions. It is stated that some of the branch data must be specified for flow in either direction!

Provided good input data can be obtained, the program is reported to give realistic results. Validation studies have been carried out on fairly simple circuits (Morse 1977).

Floyd Bossard Mine Ventilation Design System MIVENDES

This is a thermodynamic program that purports to do just about everything! It is described in Chapter 38 of Bossard (1982) which includes a comparison table of six mine ventilation programs. Floyd Bossard and Associates use this program in their mine environmental engineering consulting practice and it is not apparent whether the program is available to other users. It is clear from the description of the input data that the thermal effects are determined on the basis of rock temperatures and heat transfer coefficients, plus other sources of heat such as diesel fuel burned, electrical power used, blasting techniques, moisture evaporation and condensation, and mine air heating or cooling plants.

Additional features claimed for this system beyond the standard output for mine ventilation programs include the following:

- Barometric pressures at nodes.
- Wet- and dry-bulb temperatures.
- Various psychrometric properties associated with the above.
- Diesel exhaust gases.
- Methane emissions.
- Radon emanations.
- Separate versions are used for English and S.I. units.

Output of the calculated barometric pressure and temperature data afford a means of checking the validity of the program against measured data. This is the only program among those surveyed that incorporates radiation calculations. The program seems to rely heavily on theoretical data for input and is probably most useful for mine design exercises. The methods used for calculating radon daughter concentrations and the "age" of air are not explained.

The program appears to have one major shortcoming in that it has no provision for entering fan characteristics. It has provision for fan selection, which is easily achieved by using a fixed flow branch where the fan is required. The standard output shows the pressure needed to generate the specified flow and hence defines the operating point of a fan. However, actually matching a fan to the circuit is the ultimate test of the ability of the program to model the circuit.

The use of fixed flow branches forces a solution but gives no indication of the stability of the circuit. This is the standard procedure for selecting fans when designing a new mine or extension to an existing circuit. When the fixed flow branch is replaced by a fan characteristic curve, the balance of the circuit is affected. Depending on the stability of the rest of the circuit, the fan may or may not deliver the expected flow rate. Problems are more likely to arise where there are multiple fan installations and many paths to surface. With properly specified fan characteristics, any difficulty in reaching a solution indicates that difficulties will probably be encountered with the fan installation in the field. Any program which ignores this important step is seriously deficient.

CANMET's Thermodynamic Ventilation Network Program

This thermodynamic program (Hall 1982) features various ways of entering branch resistance data, S.I. units, network plotting, input error checking and accommodation of moisture effects. The extra data needed for the thermal and moisture effects appear to be end elevations, average air density, and moisture content for each branch.

The program is said to use a mass flow balance at each junction; however, in the sample runs illustrating the use of the program (Hall 1982), at least four solutions have been reached in spite of an obvious discrepancy in flow balance at one particular node. Besides appearing in the input data, the effect of the discrepancy is evident in each of the output tables but has not triggered any of the built-in error messages. At least, there is no mention of any error in the text.

A novel feature of this program is the removal of stalled fans from the circuit to allow the program to continue and reach a balanced solution. The purpose of this procedure is not clear since another run would be required with a substitute fan to achieve a valid solution in any case. Possibly, this technique helps to pinpoint the fan causing the problem, since a fan entering the stall region disrupts the flow distribution and may force one or more other fans into a stall condition artificially. It would then be difficult to identify the fan which precipitated the condition.

Queen's University Radiation Program

This program incorporates radon growth and decay calculations into a ventilation network analysis program (Wheeler 1982). The program is based on basic mine tunnel concepts and the physics of radon daughter ingrowth and decay (Schroeder 1969; Evans 1969).

Radon produced from a rock surface (such as the wall of an underground drift) is characterized by the emanation rate, defined as the activity of radon per unit of surface area per unit of time. The emanation rate may be measured in the field or may be estimated from laboratory measurements of emanating power and porosity. Emanating power refers to the activity of radon per unit of volume per unit of time.

Quantities of radon entering a mine atmosphere can be calculated from suitable values of these parameters together with the relevant rock surface areas and broken muck volumes.

To appreciate the growth of the radon daughter concentrations as radon-laden air moves through a mine, it is useful to consider three different radioactive components in the air passing through each branch or circuit:

- The radon daughters initially present with concentration WL_o.
- The daughters that result from the decay of the initially pure radon concentration, C_o.
- The radon daughter contribution which grows from pure radon diffusing from the rock into the mine.

Each of these components can be expressed mathematically as a function of time. The travel time of air in a given airway is the ventilation flow rate divided by the volume of the airway. Integrating the WL components over the travel time thus gives the growth in WL over the length of the airway (Schroeder 1969).

A computer program has been written at Queen's to simulate the growth of radon daughter concentrations (measured in WL) in mine airways using the above scheme with the following assumptions:

- Emanation rates are uniform over the airway surfaces.
- The activity of radon does not decrease during the period considered.
- Radon and all resulting daughter products remain suspended in the air, i.e. removal is only by ventilation or radioactive decay and not by plateout or deposition.

The input data required are:

- Airway dimensions, as well as the beginning and end nodes.
- The air flow rate in each airway.
- The emanation characteristics of each airway.

The first item forms part of the input to a standard mine ventilation network program and the second item is part of the output from the same. By providing data on the emanating characteristics therefore, the radiation calculations can be performed as a special option subsequent to a normal network solution.

Some additional features included in the program are as follows:

- It is possible to enter a number of different emanation rates and corresponding surface areas for each airway. For instance, in a seam-type deposit, the walls of a drift may be in ore while the roof and floor may be in waste. Additionally, there may be a layer of crushed ore on the floor which would have a higher emanation rate than a solid rock surface.
- Point sources of radon and radon daughters can be provided at certain nodes to simulate situations such as water-borne radiation or contamination from development headings which are ventilated by auxiliary fans and do not yet form part of the through ventilation network.
- Calculations of WL growth are applied to the airways in a particular sequence such that all flows into a given node are evaluated before the initial condition of the flows from the node can be determined.
- Where two or more airflows enter a junction their radiation levels are combined on a weighted volumetric flow average basis.

Output data consists of quantities for each airway that would be most useful for ventilation purposes.

- Final radon daughter concentration.
- Final radon concentration.
- Final equilibrium ratio.
- Final volumetric age.

SENES Ventilation Program MINEVENT

MINEVENT is a computer program written in BASIC by SENES to model radon and radon daughter concentrations in underground mine environments. It is essentially similar to the Queen's radiation program, but because it is written in BASIC, it can be used interactively at a microcomputer terminal to easily assess the effects of altering the input data. The airways are modelled as elements connected to each other by nodes. The necessary inputs for each element are: start and end node labels, cross-sectional area and perimeter, length, air flow rate, total radon flux from area sources (averaged and distributed evenly along the surface of the element) and incoming radon from sources at the entrance to the element.

The outputs are radon and radon daughter concentrations (WL) for each element, and the contribution to the total WL values from the three radioactive components modelled: the initial radon daughters present; the daughters that result from the decay of the initially pure radon at the entrance; and the radon daughters which grow from the pure radon that enters the element via area sources.

Both the Queen's model and the SENES model use simplifying formulations for the buildup and decay of daughter products. These models, based on Schroeder (1969) and Evans (1969) do not calculate specifically the concentrations of individual daughter products. Rather, an empirical equation is used to approximate the build-up and decay of the effective radon daughter concentrations measures in WL. This is a potential source of difficulty when combining air flows of different ages at a node in a ventilation network but the resultant errors are generally much smaller than the errors due to variable input data (e.g. emanation rates).

Other Models For Calculating Underground Radiation Levels

Besides the above, there are a number of theoretical models that have been developed to predict radiation levels in underground uranium mines using more detailed formulations. These have been reviewed and compared to measured values by Bigu (1981; 1983). The following discussions are taken from Bigu (1983).

The basic concepts of radon daughter decay and ingrowth were first considered by Evans (1969) and later developed in a mine ventilation model by Schroeder and Evans (Schroeder 1969). Thomas and Epps (1970) carried out similar calculations but they specifically included the growth in the concentration of individual radon daughters rather than just total WL's.

Rolle (1972) studied three models:

- a model consisting of the mixing of very "young" air, and very "old" air where equilibrium between the radon daughters had been attained;
- the "isolated" mine model in which a quantity of radon is added initially to air and then decays, there being no replenishment or removal of activity from the system; and
- the "mine-tunnel model" in which a source of radon is released at one point in the mine and then carried along with the ventilation air through the rest of the mine. Radon is constantly replenished by diffusion from the walls of the mine.

Beckman and Holub (1979) investigated a more realistic mine-tunnel model for radon which takes into account removal of decay products by ventilation.

Bigu (1983) extended the above models to include thoron and its daughters and also discussed briefly a model which takes into account the loss of daughter products by attachment to mine walls and gravitational settling.

According to Bigu (1983), these models are not highly successful, practical or even clearly distinguishable on the basis of experimental data collected in Elliot Lake mines. In fact, the simplistic models quite frequently gave better agreement than the "improved" models. The reasons for these deficiencies are the unrealistic assumptions used in the models and the oversimplification of complex dynamic situations encountered in actual practice. Bigu suggests that more realistic models would have to take into consideration the spatio-temporal dependent aspects of radon/thoron daughter removal to reflect the dynamics of the situation as well as geophysical, geochemical and geometrical considerations. The extreme complexity of the problem would require fast computers and solutions using numerical methods.

E.4 Practical Problems in Model Application

E.4.1 Model Limitations

A number of publications dealing with computer ventilation network analysis have emphasized the difficulties and importance of obtaining adequate input data (Brown 1979; McPherson 1982; Luxbacher 1982). The problems are compounded for thermodynamic programs which offer two methods of entering the psychrometric data required:

- Wet- and dry-bulb temperature and barometric pressures are entered for every node in the circuit.

  These data are often measured in a mine pressure survey in order to calculate branch resistances for an incompressible flow program so extra work is not necessarily involved. However, in a large mine, the survey process may take months and conditions in the mine may change considerably during that time. The input data should represent conditions in the mine at a particular instant in time and of course it is not possible to measure everything at once.

- Thermal gradients, heat transfer coefficients, heat and moisture sources are entered.

  These items can only be determined in very general terms. Presumably, a long and meticulous process of field measurements and adjustment of the input parameters would be required to validate the simulation. This technique lends itself more to the mine design stage than to the simulation of an existing system.

Even for an incompressible flow model, a large amount of field work is required to assemble the branch resistance data.

For the radiation program, a standard ventilation network program should be run first to demonstrate the integrity and stability of the mine ventilation system. The radiation analysis should use the flow distribution output by the network program to determine the effects of various ventilation strategies on the radiation levels.

Assuming that an acceptable flow distribution can be produced, the validity of the radiation program depends on the accuracy with which the following parameters can be assessed.

Emanation Rates

These can obviously only be measured at a finite number of locations and at specific points in time. A sufficient number of sample measurements should be taken to allow a statistical estimate to be made of the degree of confidence in the values adopted.

Perhaps values used in the program for each airway should be selected from a given distribution by a random number process. This approach might be appropriate for emanations from the floor which may be covered with variable quantities of broken muck. Alternatively, it would be possible to employ a "worst case" scenario and use the highest values measured. As a minimum requirement, it should be possible to identify a mean emanation rate for each representative rock type and ore grade.

Emanating Areas

The surface areas of mine airways can only be determined very approximately. In practical terms it is not likely that any more detailed appraisal will be possible than scaling the areas from mine plans. This technique will probably under-estimate the actual emanating area since rock surfaces underground are typically quite jagged and walls are often fractured to some degree, giving rise to a much greater potential emanating surface area.

Over a period of time, tunnel walls also become coated with layers of dust which can provide an enormous emanating surface area if the dust is derived from the ore. If the emanation rate data are determined from core samples, they would not take this effect into account. However, if they are measured directly from the rock surface underground, the effect would be included provided representative surfaces were chosen with dust layers intact. (This could cause problems in practice since it could be difficult to seal the chamber measuring the emanation rate to a rock surface covered with dust and/or mud).

Airflow Distribution

To determine the effectiveness of a ventilation strategy, the computer model must accurately simulate the flow paths underground and distribute the airflow in the same manner.

Airflow Rates

These can either be measured or generated by a computer network analysis program. In the latter case, the accuracy of the values will depend on the fidelity of the network model and the precision of the resistance values used.

Assumptions Implicit in Radiation Models

As discussed previously, a number of assumptions are made in most radiation ventilation models, not all of which are realistic. These assumptions are usually made for purposes of simplicity in order to derive solutions using analytical means or because input data necessary to implement the more complex (more realistic?) are just not available. As noted by Bigu (1983) this is particularly true for models that include radon daughter removal mechanisms other than radioactive decay, e.g. plateout.

Jacobi (1972) derived a simple model for plateout to estimate radon daughter concentrations in a turbulent mixed air volume in steady-state equilibrium taking into account attachment fractions and removal by ventilation and plateout (a room, not a mine model). Porstendorfer (1978) also studied the influence of removal processes on the concentration of radon daughters in room air. Measurements by George (1983) and modelling by Knutson (1983) were performed to optimize the Jacobi model. These researches point out the difficulties in attempting to model radon daughter removal mechanisms, even under well-defined conditions, e.g. particle size distributions and particle densities must be considered.

E.4.2 Other Factors Affecting Model Predictions of a Mine Ventilation System

In addition to the above noted difficulties, there are a number of other factors that could upset or unbalance a mine ventilation system and hence affect the utility of model prediction. These factors are inherent in the operation of any underground mine but because of their random nature cannot be specifically accounted for in advance in a mine ventilation model (although they might be modelled as specific "what if" scenarios to determine their potential effects). These factors include:

- unauthorized change of a ventilation door (from closed to open or vice-versa)
- damage to or increased leakage of air through a formerly tight ventilation bulkhead in a drift, raise, boxhole, manway, etc.
- the breaking through of a raise or drift or stope into another mine working may cause an upset to the ventilation system unless proper precautions have been taken to prevent such an upset. The upset may include decrease or increase of flows or reversal of flows in certain areas of the ventilation network.
- failures of pillars in abandoned workings may have the same effect unless anticipated and the upset is minimized by the installation of appropriate bulkheads and doors.

- failure of a main or a booster fan because of mechanical or electrical faults. This may cause partial or total failure of the ventilation system depending on the intricacies of the system and the percentage of the total ventilating fans that failed.
- the passage of ore trains through long drifts and cross-cuts can accelerate or impede or reverse normal air flows, especially when the train occupies a large percentage of the cross-sectional area of the heading (the piston effect).
- the parking of ore trains or the piling ot supplies in airways can effectively throttle the flow of air.
- undetected falls-of-ground in airways through infrequented mine areas, or falls-of-ground that are not cleaned up can effectively throttle the normal flow of air. In northern areas the build-up of ice in supply airways near surface may have the same effect. Also the installation of timberings where there is bad ground can also restrict the effective area of an airway and reduce air flows.
- a damaged bulkhead or crushed pillar that permits leakage of radon gas from an abandoned mining area can adversely effect the quality of the mine air, though the actual volume flow is changed little.
- an upset in the mine ventilation system due to a damaged bulkhead or the unauthorized change in a ventilation door can cause recirculation of air containing radon. The recirculation will increase the "age of the air" before it reaches the working places and because of this increase in age, the radon daughter levels may become unacceptable.
- mine fires can drastically affect mine ventilation networks by the release of toxic gases, and because of the heat released, some air flows may be stopped or reversed.
- the operation of ventilation fans on exhaust raises from deep mines may be adversely affected by humidity

condensing in the upcast saturated air, or by water seeping into the shaft. Air velocity should be kept above or below certain velocities in such vertical shafts. Otherwise the exhaust volume of such fans may be very irregular as they go into and out of the stall position as the water load gathers in or falls out of the air streams.

REFERENCES FOR APPENDIX E

Beckman, R.T. and R.F. Holub, 1979. "Radon Daughter Growth with Continuous Radon Influx and Various Ventilation Rates." U.S. Dept. of Labour, MSHA Investigation Report, IR 1099.

Bigu, J., 1981. "Mine Models and the Thoron Problem in Underground Uranium Mines." Proc. Int. Conf. on Radiation Hazards in Mining - Control Measurement and Medical Aspects, Golden Colorado, pp 243-255, SME-AIME, New York, Manual Gomez (Ed.), 5-9 Oct.

Bigu, J., 1983. "Theoretical Models for Determining Radon and Thoron Daughter Levels in Canadian Underground Uranium Mines - A Comparison With Experimental Data." CANMET: Energy, Mines and Resources Canada, Divsion Report MRP/MRL 83-149 (J), (revised).

Bossard, F.C., J.J. Lefever, J.B. Lefever, and K.S. Stout, 1982. "A Manual of Mine Ventilation Design Practices." 2nd Ed. Chap. 38, Floyd Bossard & Associates, Butte, Montana.

Brown, J.R., T.H. Fisher, and K.L. Gibson, 1979. "The Development, Application and Correlation of Computer Simulation Techniques of the Mt. Isa Ventilation System." 2nd International Mine Ventilation Congress, Reno, Nevada.

Burrows, J., R. Hemp, W. Holding and R.M. Stroh (eds.), 1982. "Environmental Engineering in South African Mines." Mine Ventilation Society of South Africa.

Coulter, M., 1984. Engineering Department, INCO, Sudbury. Personal Communication, February.

Evans, R.D., 1969. "Engineers' Guide to the Elementary Behaviour of Radon Daughters." Health Physics, Vol. 17, p. 229.

George, A.C., E.O. Knutson and K.W. Tu, 1983. "Radon Daughter Plateout-I. Measurements." Health Physics, Vol. 45, p.439.

Greuer, R.E., 1979. "A New Computer Program for the Design of Ventilation Emergency Plans." 2nd International Mine Ventilation Congress, Reno, Nevada.

Hall, A.E., M.A. Stoakes, and M.K. Gangal, 1982. "CANMET's Thermodynamic Ventilation Network Program." CIM Bulletin, December.

Hartman, H.L., 1982. "Mine Ventilation and Air Conditioning." (Second Edition) (Co-editors: Mutmansky, J.M., Wang, Y.L.), John Wiley and Sons, Inc.

Jacobi, W., 1972. "Activity and Potential Energy of Rn-222 and Rn-220 Daughters in Different Air Atmospheres." Health Physics, Vol. 22, p.441.

Knutson, E.O., A.C. George, J.J. Frey and B.R. Koh, 1983. "Radon Daughter Plateout-II. Prediction Model." Health Physics, Vol. 45, p.445.

Luxbacher, G.W. and R.J. Ramani, 1982. "Developing Input Data for Computer Simulation of Mine Ventilation Systems from a Pressure-Quantity Survey." 1st Mine Ventilation Symposium, SME AIME, University of Alabama.

McPherson, M.J. and M. Hood, 1982. "Sunnyside No. 3. A Case Study in Ventilation Planning." 1st Mine Ventilation Symposium,SME AIME, University of Alabama.

Morse, R.L., 1977. "The Use of a Thermodynamic Ventilation Program." SEM AIME Fall Meeting, St. Louis, Mo.

Porstendorfer, J., A. Wiche and A. Schraub, 1978. "The Influence of Exhalation, Ventilation and Deposition Processes Upon the Concentration of Radon and Thoron and Their Decay Products in Room Air." Health Physics, Vol. 34, p.465.

Ramani, R.V., A. Owili-Eger, and C.B. Manula, 1975. "A Master Environmental Control and Mine System Design Simulator for Underground Coal Mining." Vol. X Ventilation Subsystem, Penn. State University.

Rolle, R., 1972. "Radon Daughters and Age of Ventilation Air." Health Physics, Vol. 23, p.128.

Schroeder, G.L. and R.D. Evans, 1969. "Some Basic Concepts in Uranium Mine Ventilation." Trans. SME AIME Vol. 244, p.301.

Stachulak, J., 1979. "Computer Network Calculation of Creighton Mine Mass Flow and Natural Ventilation." 2nd International Mine Ventilation Congress, Reno, Nevada.

Thomas, J.W. and J.W. Epps, 1970. "Radium-A, -B and -C Concentrations And Working Levels from Radon Emanation in a Mine Tunnel." HASL-TM 70-4.

Wang, Y.J., and L.W. Saperstein, 1970. "Computer Aided Solution of Complex Ventilation Networks." SME AIME, Vol. 247, p. 238.

Wheeler, A.J., 1982. "The Incorporation of Radon Growth Calculations Into Computer Ventilation Network Analysis." M.Sc. Thesis, Mining Engineering Dept., Queen's University, Kingston, Ont.